



Universidad
Carlos III de Madrid

Departamento de Teoría de la Señal y de la Comunicación

PROYECTO FIN DE CARRERA

Técnicas de Detección para Aplicaciones de Localización en Redes de Sensores

Autor: Marta Salas Costumero

Tutor: Sara Pino Povedano

Leganés, 12 diciembre de 2011

Título: Técnicas de Detección para Aplicaciones de Localización en Redes de Sensores.
Autor: Marta Salas Costumero
Director: Sara Pino Povedano

EL TRIBUNAL

Presidente: Francisco Javier González Serrano

Vocal: Irene Pérez Encinar

Secretario: Matilde Sánchez Fernández

Realizado el acto de defensa y lectura del Proyecto Fin de Carrera el día 12 de diciembre de 2011 en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de

VOCAL

SECRETARIO

PRESIDENTE

Agradecimientos

A mi tutora Sara, porque me ha ayudado a que este proyecto haya podido salir adelante. Me ha resuelto los problemas cuando me quedaba atascada en algún sitio, y me ha animado a tener valentía y fuerzas para conseguir terminarlo. Agradecer todo el tiempo que ha empleado para que este proyecto saliera adelante.

A mi familia, en especial a mis padres y a mi hermana, que han sido los que han tenido que soportar todo mi estrés, mis malas contestaciones y siempre me han animado a continuar hacia delante y a que en los momentos de dificultad consiguiera reponerme. Si no hubiera sido por todo su apoyo, y por los esfuerzos que han hecho económicamente, yo no sé si hubiera sido capaz de terminar la carrera. También nombrar al resto de mi familia que siempre se ha preocupado en preguntarme cómo iba y en valorar todo mi esfuerzo.

A mis compañeros de la universidad. A Andrés por haberme tolerado mis gritos cuando veía que no nos salían las cosas. A Patri por sus largas conversaciones en la época de exámenes para cerciorarme de que tenía los conocimientos suficientes para el examen. A María, Diana, Estefanía, Elena, Jaime, Dani, Rober, Jose, Álvaro, Varuko, Miriam, Irene, Marta, M^a Ángeles, Sergio, Fernando, Barco, Xavi, Almu, Isma, Javi, Jonny, y alguno más que ha pasado por mi vida universitaria, que siempre los he tenido ahí para cualquier duda, para desconectar en las comidas, para pasar un buen rato, para dejarnos apuntes, para lo que sea. A todos vosotros gracias, que durante todo este tiempo habéis sido grandes compañeros y he pasado más tiempo con vosotros que mi familia.

A mis amigos del barrio que siempre han estado ahí y me han apoyado y animado cuando el estrés se apoderaba de mí. Los cuales siempre se han preocupado de saber cómo iba y de animarme en los momentos difíciles.

A alguna gente del pueblo que también me han animado a salir adelante, en concreto a Alberto, que en los momentos difíciles estaba ahí para animarme, y decirme las cosas que necesitaba oír, su ayuda también ha sido importante para poder salir adelante.

En general gracias a todos los que habéis estado a mi alrededor durante este periodo de tiempo, sin vosotros no hubiera podido!!!!

Resumen

Las redes de sensores inalámbricos se han convertido en un campo de investigación muy importante en los últimos años. Esta clase de redes se caracterizan por su facilidad de despliegue y por ser autoconfigurables, pudiendo convertirse en todo momento en emisor, receptor, ofrecer servicios de encaminamiento entre nodos sin visión directa, así como registrar datos referentes a los sensores locales de cada nodo. Otra de sus características es su gestión eficiente de la energía, que les permite obtener una alta tasa de autonomía que las hacen plenamente operativas.

El seguimiento y localización de objetivos es una de las aplicaciones más importantes de las redes de sensores. Estas redes son ideales para el seguimiento, debido a su gran cobertura, y a su multiplicidad sensorial y modal. Dicha capacidad de seguimiento se basa en la información local adquirida por cada sensor, y en el conocimiento de su localización.

En este proyecto se analizarán e implementarán distintas técnicas de detección y un algoritmo de localización. En este análisis se van a distinguir dos tipos de procesamiento de red y dos tipos de despliegue de la misma, para comparar el funcionamiento de las técnicas en esos entornos. También se hará una breve introducción a lo que es una red de sensores. Por último se analizarán los resultados que se han obtenido al implementar dichas técnicas para obtener las conclusiones oportunas.

Palabras clave: redes de sensores, centralizado, distribuido, CFAR, Decision Fusion Rule, filtro de partículas, umbral, nodo.

Abstract

Wireless sensors networks have been become a very important research field in last years. This kind of networks are characterized by their facility of deployment and to be self-configuring, it can be become all time transmitter, receiver, provide routing services between nodes without direct vision, as well as record data on local sensor of each node. Another feature is its efficiency energy management, allowing them to obtain a high level of autonomy that it is done them fully operational.

Tracking and tracing objectives is one of the most important applications of sensor networks. These networks are ideal for monitoring, due to its wide coverage, and its multiplicity sensory and modal. This tracking capability is based on local information acquired by each sensor, and knowledge of their location.

In this project it will be analyzed and implemented several techniques for detection and localization algorithm. In this analysis it will be distinguished two types of network processing and two types of deployment of it, to compare the performance of the techniques in such settings. Also it will be provided a brief introduction to what is a network of sensors. Finally, it will be analyzed the results have been obtained by implementing these techniques to obtain appropriate conclusions.

Keywords: sensor networks, centralized, distributed, CFAR, Decision Fusion Rule, particulate filter, threshold, node.

Índice general

CAPÍTULO 1	18
INTRODUCCIÓN	18
1.1 REDES DE SENSORES	18
1.1.1 Componentes de una red de sensores.....	19
1.1.2 Aplicaciones	20
1.1.3 Detección y Localización en redes de sensores	23
1.2 OBJETIVOS DEL PROYECTO	24
1.3 ESTRUCTURA DE LA MEMORIA	24
CAPÍTULO 2	27
ESCENARIO DE TRABAJO.....	27
2.1 PROCESAMIENTO EN REDES DE SENSORES.....	27
2.1.1 Procesamiento Centralizado	28
2.1.2 Procesamiento Distribuido.....	30
2.2 DESPLIEGUE DE LA RED	32
2.2.1 Despliegue aleatorio de sensores.....	32
2.2.2 Distribución de los sensores en forma de rejilla	33
2.3 LOCALIZACIÓN DE BLANCOS	34
2.3.1 Modelo de movimiento del blanco.....	35
2.4 MODELO DE MEDIDA	36
CAPÍTULO 3	39
DETECCIÓN EN REDES DE SENSORES.....	39
3.1 DETECCIÓN DE SEÑALES.....	39
3.1.1 Formulación	40
3.2 DETECCIÓN EN REDES DE SENSORES	41
3.3 TÉCNICAS DE DETECCIÓN	43
3.3.1 Constant False Alarm Rate Detector.....	43
3.3.2 Decision Fusion Rule.....	45
3.3.2.1 Optimal Decision Fusion Rule	46
3.3.2.2 Aproximación de Optimal Decision Fusion Rule	46

CAPÍTULO 4	48
ALGORITMO DE LOCALIZACIÓN	48
4.1 FILTROS DE PARTÍCULAS	48
4.1.1 <i>Formulación del filtro de partículas</i>	49
4.1.2 <i>Remuestreo</i>	52
4.1.3 <i>Aproximación de Bootstrap</i>	54
CAPÍTULO 5	57
RESULTADOS EXPERIMENTALES	57
5.1 ELECCIÓN DE PARÁMETROS	57
5.1.1 <i>Elección de parámetros para CFAR</i>	57
5.1.2 <i>Elección de Parámetros para Fusion Decision Rule</i>	61
5.2 PRESTACIONES	61
5.2.1 <i>Números medio de vecinos</i>	62
5.2.2 <i>Probabilidad de falsa alarma y de detección</i>	63
5.2.3 <i>Número de sensores activos en función del ruido</i>	64
5.2.4 <i>Prestaciones en Localización</i>	68
CAPÍTULO 6	74
CONCLUSIONES	74
6.1 CONCLUSIONES	74
6.2 FUTURAS LÍNEAS DE TRABAJO	76
CAPÍTULO 7	78
PRESUPUESTO	78
7.1 PLANIFICACIÓN	78
7.2 PRESUPUESTO	80
7.2.1 <i>Coste del personal</i>	80
7.2.2 <i>Coste del hardware</i>	80
7.2.3 <i>Coste del software</i>	81
7.2.4 <i>Coste del Material de Oficina</i>	82
7.2.5 <i>Costes indirectos</i>	82
7.2.6 <i>Coste Total</i>	82
REFERENCIAS	86

Índice de figuras

Figura 1 : Esquema detallado de lo que contiene cada dispositivo. [7].	18
Figura 2 : Funcionamiento de la red. [7].	20
Figura 3 : Ejemplo de una red con procesamiento centralizado.	28
Figura 4 : Ejemplo de una red con procesamiento distribuido. [18]	30
Figura 5: Ejemplo de la distribución de sensores siguiendo una variable aleatoria uniforme.	33
Figura 6: Distribución de los sensores en forma de rejilla cubriendo completamente el área de estudio.	34
Figura 7 : Representación de siete trayectorias generadas con el modelo de movimiento.	36
Figura 8 : Potencia recibida por el sensor en función de la distancia ($a = 20$ dB y $\alpha = 0.25$).	37
Figura 9 : Representación de la función densidad de probabilidad.	42
Figura 10: Ejemplo de partículas y coeficientes para la aproximación de una distribución de probabilidad.	50
Figura 11 : Ejemplo de convergencia de un filtro de partículas.	52
Figura 12 : Ejemplo de remuestreo de un filtro de partículas.	53
Figura 13 : Ejemplo de reconfiguración de las partículas en el remuestreo.	54
Figura 14 : Resultado para un valor de $\alpha = 0.3$.	59
Figura 15 : Resultado para un valor de $\alpha = 0.5$	59
Figura 16 : Resultado para un valor de $\alpha = 0.9$	60
Figura 17 : Número de sensores activos medio en una red centralizada con despliegue en rejilla.	65
Figura 18 : Número de sensores activos medio para una red centralizada con despliegue aleatorio.	66
Figura 19 : Número de sensores activos medio para una red distribuida con despliegue en rejilla.	67
Figura 20 : Número de sensores activos medio para una red distribuida con despliegue aleatorio.	68

Figura 21 : RMSE obtenido para una red centralizada con despliegue de rejilla.....	69
Figura 22 : RMSE para una red centralizada con despliegue aleatorio.....	70
Figura 23 : RMSE para una red distribuida con despliegue en forma de rejilla.....	71
Figura 24 : RMSE para una red distribuida con despliegue aleatorio.	72
Figura 25 : Presupuesto total.	83

Índice de tablas

Tabla 1 : Representación de los posibles estados de las decisiones tomadas.....	40
Tabla 2 : Número medio de vecinos promediado con varias iteraciones.	62
Tabla 3 : Probabilidad de falsa alarma y de detección para una red centralizada.	63
Tabla 4 : Probabilidad de falsa alarma y de detección para una red distribuida.	64
Tabla 5 : Planificación del proyecto. Especificación de la duración de sus fases.	79
Tabla 6 : Coste del personal detallado.....	80
Tabla 7 : Coste detallado del material hardware.	81
Tabla 8 : Coste detallado del material software.	81

Capítulo 1

Introducción

1.1 Redes de Sensores

Una red de sensores inalámbricos (*Wireless Sensor Network - WSN*) es un sistema compuesto por numerosos dispositivos que son distribuidos espacialmente y utilizan sensores para controlar diversas condiciones ambientales, como por ejemplo la temperatura, el sonido, la vibración, la presión, el movimiento o los contaminantes.

Los dispositivos que componen esta red son unidades autónomas que constan de un microcontrolador, una fuente de energía (casi siempre una batería), un radio transceptor y un elemento sensor. En la Figura 1 se puede ver un ejemplo de cada uno de los bloques que componen estos dispositivos. Los sensores tienen ciertas capacidades sensitivas y de comunicación que permiten formar redes inalámbricas Ad-Hoc sin infraestructura física preestablecida ni administración central.



Figura 1 : Esquema detallado de lo que contiene cada dispositivo. [7].

Debido a las limitaciones de la vida de la batería, los nodos se construyen teniendo presente la conservación de la energía, y generalmente pasan mucho tiempo en modo 'durmiente' (sleep) de bajo consumo de potencia. Los nodos autoorganizan sus redes en una forma ad hoc, en lugar de tener una topología de red previamente programada. Además, las redes de sensores tienen la capacidad de autorrestauración, es decir, si se avería un nodo, la red encontrará nuevas vías para encaminar los paquetes de datos. De esta forma, la red sobrevivirá en su conjunto, aunque haya nodos individuales que pierdan potencia o se destruyan.

Esta clase de redes se caracterizan por su facilidad de despliegue y por ser autoconfigurables, pudiendo convertirse en todo momento en emisor, receptor, ofrecer servicios de encaminamiento entre nodos sin visión directa, así como registrar datos referentes a los sensores locales de cada nodo. Otra de sus características es su gestión eficiente de la energía, con la que consiguen una alta tasa de autonomía que las hace plenamente operativas. Cada nodo, como ente individual de una red de sensores, no deja de ser una pequeña computadora, con un pequeño procesador, una memoria de programa y una memoria para almacenar variables, pero al que también se agregan unos pequeños periféricos I/O (entrada/salida) tales como un transceptor radio y un pequeño conversor A/D (Análogo/Digital) que sirve para adquisición de los datos de los sensores locales.

1.1.1 Componentes de una red de sensores

En su forma más simple, una red de sensores inalámbricos puede basarse en una estación base en la que se realiza la recogida de datos que se han tomado en todos los otros dispositivos (nodos) de una manera sincronizada y estructurada. La funcionalidad de la red, es decir, las rutas y protocolos de comunicación, se implementan en software y se ejecutan generalmente en microcontroladores de gama baja.

➤ Sensores y actuadores

Un sensor es un dispositivo físico que detecta propiedades físicas, biológicas, y/o químicas de su ambiente y convierte estas propiedades en una señal eléctrica. Un actuador normalmente acepta una señal eléctrica y la convierte en una acción física para actuar sobre el medio en el que se encuentra.

Por otro lado, cada nodo sensor soporta un algoritmo de enrutado multisalto, de forma que dentro de una red de sensores inalámbrica, los nodos se autoorganizan y colaboran en la detección y el reenvío de información hacia uno o varios destinos. Además, los nodos de una red de este tipo deben ser robustos para permitir su implantación en medios hostiles, deben tener un bajo consumo energético y su producción ha de ser económica.

➤ Transceptor

Cada uno de los nodos de un sistema de redes de sensores inalámbricos comprende una unidad de transceptor, que está a cargo de la comunicación inalámbrica con sus compañeros.

➤ Un microcontrolador

Un microcontrolador es un ordenador en un chip, autosuficiente y de bajo costo que se utiliza para controlar la funcionalidad y el flujo de datos de dispositivos electrónicos para almacenar y/o procesar datos.

De manera conjunta la red funciona de la siguiente manera: los dispositivos sensores detectan un cambio en la magnitud física que están midiendo, esa información la transmiten al transceptor de cada nodo, quien a su vez envía la información al microcontrolador el cual contiene el programa de la red y es el que se encarga de decidir que sucede en la red. Una vez se procese esta información, el microcontrolador enviará la señal de respuesta al transceptor de cada nodo y éste a los dispositivos últimos que son los sensores y/o actuadores (Figura 2).

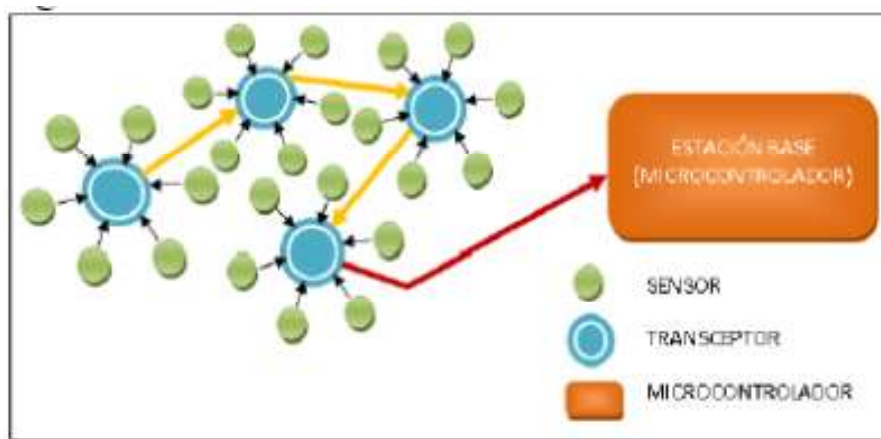


Figura 2 : Funcionamiento de la red. [7].

➤ Administración de energía

Existen técnicas para hacer buen uso de la energía y obtenerla a través de diferentes fuentes además de las baterías. La primera es la transformación de las vibraciones en electricidad a través de efectos de inducción piezoeléctricos, electrostáticos y magnéticos, con la cual se puede proporcionar hasta $300\mu\text{W}/\text{cm}^3$. La segunda es la transformación de un flujo de calor en electricidad y puede proporcionar hasta $40\mu\text{W}/\text{cm}^3$. La tercera recupera la energía de la luz - a niveles de hasta $15000\mu\text{W}/\text{cm}^3$ al aire libre y de interior a $10\mu\text{W}/\text{cm}^3$ - o de la radiación de RF (Radio Frecuency) de hasta decenas de milivatios en el caso de una fuerte fuente como un lector de etiquetas RFID (Radio Frecuency IDentification).

1.1.2 Aplicaciones

Las redes de sensores pueden estar compuestas por diferentes tipos de sensores como son termales, visuales, acústicos, de radar, de baja tasa de muestreo magnética, etc., los cuales son capaces de monitorizar una gran variedad de condiciones en el entorno, incluyendo:

- Temperatura
- Humedad
- Movimiento de vehículos
- Estado de relámpagos
- Presión
- Niveles de ruido
- Presencia o ausencia de cierto tipo de objetos
- Composición del suelo
- Niveles de carga mecánica en objetos unidos, y
- Características actuales como son velocidad, dirección y tamaño de un objeto.

Los nodos sensores pueden ser usados para sensado continuo, detección, identificación, localización del sensado y para el control local de los actuadores. El concepto de micro-sensado y comunicación inalámbrica de estos sensores promete muchas nuevas áreas de aplicación. Se va a categorizar las aplicaciones en militares, ambientales, de la salud, del hogar y otras áreas comerciales. Es posible expandir esta clasificación con más categorías como son la exploración espacial, procesamiento químico y desastres del relieve.

- Aplicaciones militares:

Las redes de sensores inalámbricas pueden ser una parte integral del comando, control, comunicaciones, cómputo, inteligencia, supervivencia, reconocimiento y sistemas de localización militar (C4ISRT (command, control, communications, computing, intelligence, surveillance, reconnaissance and targeting)). El rápido desarrollo, la propia organización y las características de falta de tolerancia de las redes de sensores pueden hacerlas una técnica de medida muy prometedora para los sistemas C4ISRT militares. Desde que las redes de sensores están basadas en el denso desarrollo de los nodos de sensores desechables y de bajo coste, la destrucción de algunos nodos por acciones hostiles no afectan a una operación militar tanto como la destrucción de un sensor tradicional, que hace del concepto de redes de sensores un mejor aprovechamiento de los campos de batalla. Algunas de las aplicaciones de las redes de sensores son monitorizar fuerzas amistosas, equipamiento y munición; vigilancia del campo de batalla; reconocimiento de fuerzas opuestas y terreno; seguimiento; tasación de los daños de la batalla; y detección y reconocimiento de ataques nuclear, biológico y químico (NBC).

- Aplicaciones ambientales:

Algunas aplicaciones ambientales de las redes de sensores incluyen el seguimiento de los movimientos de los pájaros, pequeños animales, e insectos; monitorizar las condiciones ambientales que afectan a las cosechas y al ganado; riego; macroinstrumentos de gran escala para monitorizar la Tierra y exploración planetaria; detección química/biológica; precisión de la agricultura; biológico, Tierra, y monitorización ambiental en mar, suelo, y contextos atmosféricos; detección de fuegos forestales; investigación meteorológica y geográfica; detección de inundaciones; y estudio de la contaminación [11,14]

CAPÍTULO 1 INTRODUCCIÓN

- Aplicaciones de la salud:

Algunas de las aplicaciones de la salud de las redes de sensores son proveer interfaces para los minusválidos; monitorizado de pacientes integrados; diagnóstico; administración de medicamentos en los hospitales; monitorizado de movimientos y procesos internos de insectos y otros pequeños animales; telemonitorización de datos psicológicos de humanos; y seguimiento y monitorización de doctores y pacientes dentro de los hospitales [8,20].

- Aplicaciones del hogar:

Automatización del hogar: Como la tecnología avanza, los nodos sensores pequeños y los actuadores pueden ser integrados en electrodomésticos, tales como aspiradoras, microondas, neveras, y vídeos [21]. Estos nodos sensores dentro de los dispositivos domésticos pueden interactuar entre ellos y con la red externa mediante Internet o por vía Satélite. Ellos permiten a los usuarios finales manejar los dispositivos del hogar de forma local o de forma remota de una manera más sencilla.

Entorno elegante: El diseño de un entorno elegante puede tener dos perspectivas diferentes, es decir, centrada en el ser humano o centrada en la tecnología [1]. Para la perspectiva centrada en el ser humano, un entorno elegante tiene que adaptarse a las necesidades de los usuarios finales en términos de capacidades de entrada/salida. Para la perspectiva centrada en la tecnología, las nuevas tecnologías hardware, las soluciones de red, y servicios middleware tienen que ser desarrollados. Un escenario de cómo los nodos sensores se pueden usar para crear un entorno elegante se describe en [16]. Los nodos sensores pueden ser integrados en el mobiliario y en los electrodomésticos, y ellos pueden comunicarse entre ellos y con el servidor de la habitación. El servidor de la habitación sólo puede comunicarse con otros servidores del resto de habitaciones para aprender sobre los servicios que él ofrece, por ejemplo, pintar, escanear y enviar fax. Estos servidores de habitación y los nodos sensores pueden integrarse con dispositivos integrados ya existentes para llegar a ser propiamente organizados, propiamente regulados y adaptarse a los sistemas basados en modelos de teoría de control como se describe en [16]. Otro ejemplo de entorno elegante es el “Laboratorio Residencial” en el Instituto de Tecnología de Georgia [10]. El cómputo y la medida en este entorno tienen que ser realizables, persistentes y transparentes.

- Otras aplicaciones comerciales:

Algunas de las aplicaciones comerciales monitorizan la fatiga del material; construyen teclados virtuales; dirigen inventarios; monitorizan la calidad del producto; construyen elegantes espacios de oficina; controlan el entorno en un edificio de oficinas; controlan robot y asesoran en entornos de fabricación automática; juegos interactivos; museos interactivos; control y automatización del proceso de fábrica; monitorización de los áreas de desastre; estructuras elegantes con nodos sensores introducidos en su interior; diagnósticos de máquinas; transportación; instrumentación de fábrica; control local de actuadores; detección y monitorización de robos de coches; seguimiento y detección de vehículos; e instrumentalización de cámaras de procesamiento de semiconductor, maquinaria rotatoria, túneles de viento, y cámaras anecóicas [15,17].

1.1.3 Detección y Localización en redes de sensores

A lo largo del apartado anterior se ha podido comprobar la versatilidad que ofrece este tipo de redes, pudiendo encontrarse en muy distintos ámbitos (ya sean escenarios u aplicaciones en entornos cerrados o al aire libre) o con distintos objetivos (aplicaciones de tipo comercial, militar, etc.). A pesar de esta variedad, existe un rasgo común a la mayor parte de las aplicaciones basadas en redes de sensores: la detección de eventos.

Las redes de sensores, con sus capacidades de medida de parámetros físicos, tales como movimiento, sonido, potencia o temperatura, permiten la detección de eventos de cualquier tipo, proporcionando así información al usuario sobre cuándo se producen situaciones anómalas (como por ejemplo la detección de incendios o vibraciones elevadas en edificios, puentes, estructuras, etc.) o relevantes (como la detección de animales en entornos protegidos), incluso a veces, de situaciones que no pueden ser detectadas por el ser humano a simple vista. En todos estos casos será necesario distinguir entre los casos importantes, en los que se está produciendo el evento que interesa monitorizar, y todos aquellos casos rutinarios que no aportan información alguna al respecto. Debido a ello se puede considerar que la detección es una de las tareas más importantes para cualquier aplicación de redes de sensores.

Este proyecto se va a centrar en el análisis de distintas técnicas de detección, principalmente en aquellas relacionadas con la localización de blancos en movimiento. En la literatura actual se pueden encontrar una gran variedad de artículos relacionados con este tipo de aplicaciones de localización.

La detección dentro de este ámbito se define como una detección de señales: el objetivo a localizar emite una señal de cualquier tipo que será captada por los sensores que componen la red, que a su vez, serán los encargados de identificar si la señal captada es solo ruido o tiene una componente de señal. Resulta evidente la importancia de una buena identificación de estos casos, puesto que identificar como señal algo que en realidad es ruido llevaría a un funcionamiento completamente erróneo de una aplicación de localización.

Cabe destacar que la detección en este ámbito ayuda a que el ser humano tenga un conocimiento mayor y más exhaustivo de todo lo que le rodea, además puede ayudarle en su vida, ya que si por ejemplo, en una fábrica, se detectan mejor los problemas que haya en el material que se está fabricando, entonces los objetos que se hagan con ese material serán de mayor calidad y por lo tanto tendrá una vida mayor, por lo que será bueno para el ser humano.

En el caso que se está estudiando, la detección va unida con el seguimiento y la determinación de la trayectoria que sigue el objetivo que se ha detectado. En el caso de este estudio, ese objetivo emite sonidos, pero se podría ampliar a algo visual, con la ayuda incluso de una cámara. La importancia de detectar una trayectoria puede ser muy grande, dependiendo del objetivo para el que se use. Por ejemplo en una tienda, como ya se ha mencionado anteriormente, se puede estudiar la trayectoria que siguen los consumidores, y se pueden ver cuáles son los pasillos más visitados, cuáles son dónde están más tiempo, etc. Esto puede servir, para hacer una reubicación de los productos y así poner los objetos con los que se saca mayor beneficio en los pasillos más visitados, esto produciría un aumento en el capital de la tienda.

Otro ejemplo, podría ser integrar los sensores en un sistema de seguridad, para poder determinar el recorrido que hacen los ladrones o los posibles ladrones, y así poder algún tipo de mecanismo que impida que los futuros ladrones se cuelen por el mismo sitio y con la misma facilidad. Hay otros muchos ejemplos que se podrían mencionar, pero lo importante es destacar que con la detección de una simple trayectoria se puede obtener mucha información que a simple vista no se puede observar. Si a esa detección de la trayectoria se le pueden añadir otros factores de tiempo, espacio, lugar y movimiento, entonces la información que se obtiene es mucho más productiva, y se pueden llegar a obtener determinados comportamientos que están relacionados.

Cada vez más la detección está presente en la vida del ser humano y cada vez surgen más investigaciones alrededor de ella, y esto se debe a que todo el mundo se está dando cuenta de que algo tan sencillo como detectar determinadas variables, pueden llegar a mejorar la calidad de vida del ser humano y de cualquier ser vivo que habita en el planeta.

1.2 Objetivos del proyecto

El objetivo de este proyecto es el análisis y comparación de distintas técnicas de detección para redes de sensores. De forma más detallada, los objetivos que se deben cumplir son:

- Estudio de las distintas técnicas de detección.
- Aplicación de algoritmos de localización sobre dichas técnicas.
- Comparación de los resultados obtenidos.
- Implementación de todo lo anterior para diversos tipos de procesamiento de datos (centralizado y distribuido) y distribuciones de sensores (posicionamiento aleatorio y determinista).

1.3 Estructura de la memoria

La memoria se ha estructurado de la siguiente forma:

Capítulo 1. Introducción \Rightarrow En este capítulo se explica en qué consiste este proyecto, así como la importancia del desarrollo del mismo dando razones de por qué se ha elegido y cuáles son sus objetivos. Situando al lector en el contexto del proyecto y así vaya comprendiendo el desarrollo del mismo.

Capítulo 2. Escenario de Trabajo \Rightarrow En este capítulo se explica el escenario en el que se ha desarrollado este proyecto. Se explicarán las distintas distribuciones de los sensores que se han implementado, así como las dimensiones de dicho escenario. También se mencionará el tipo de sensor que se ha utilizado así como su modelo de medida. Además se hará un análisis del objetivo que se va a analizar, determinando cuál será el modelo de movimiento que seguirá.

Capítulo 3. Detección en Redes de Sensores \Rightarrow A lo largo de este capítulo se explicarán las distintas técnicas que se han comparado para llegar a unas determinadas conclusiones. Además de hacer el desarrollo teórico se hablará de cómo se obtienen los parámetros característicos de una forma general.

Capítulo 4. Algoritmo de Localización \Rightarrow En este capítulo se explica el algoritmo que se ha utilizado para estimar la trayectoria que sigue el objetivo a seguir. Se dará también la razón por la que se ha elegido dicho algoritmo, ya que de él dependen gran parte de los resultados de la estimación realizada.

Capítulo 5. Resultados Experimentales \Rightarrow En este capítulo se exponen los resultados que se han obtenido al aplicar las diferentes técnicas de detección en el escenario propuesto, así como de implementar el algoritmo de localización elegido. Se van a comparar los resultados teniendo en cuenta las distintas formas de despliegue de los sensores y los distintos tipos de procesamiento que se han explicado en los capítulos anteriores.

Capítulo 6. Conclusiones y Líneas Futuras \Rightarrow En este último capítulo se expondrán las conclusiones que se han obtenido después del estudio y además se darán pequeños indicios de por dónde se podría continuar estudiando y analizando para obtener mejoras en el futuro.

Capítulo 7. Presupuesto \Rightarrow En este capítulo se va a realizar el presupuesto del análisis que se ha realizado, explicando todos los costes y argumentando su utilidad dentro de dicho proyecto.

Referencias \Rightarrow Se nombra la bibliografía que se ha utilizado para desarrollar este estudio por si se quisiera ampliar información sobre alguno de los puntos que aquí se resumen.

Capítulo 2

Escenario de Trabajo

A largo de este capítulo se presentará el escenario en el que se ha desarrollado este proyecto. Concretamente se explicarán los distintos tipos de procesamiento existentes para las redes de sensores: procesamiento centralizado y distribuido. Así mismo se detallarán los distintos despliegues para la red de sensores que se han tenido en cuenta. Después se hará un estudio del blanco que se ha elegido para seguir, explicando cuál es su modelo de movimiento. Por último se hará una explicación del modelo de medida utilizado y de cuáles podrían haberse utilizado.

2.1 Procesamiento en redes de sensores

Tal y como se ha podido comprobar en el Apartado 1.1.3, muchas de las aplicaciones de las redes de sensores requieren el procesamiento y la combinación de los datos proporcionados por varios sensores. Básicamente se pueden distinguir dos tipos de procesamiento para este tipo de redes: el procesamiento centralizado y el distribuido o colaborativo. A lo largo de este proyecto se analizarán e implementarán ambos tipos de procesamiento.

2.1.1 Procesamiento Centralizado

Los sistemas de procesamiento centralizado se basan en la utilización de una estación base, situada en uno de los extremos de la red, que será la encargada de recopilar y procesar todas las medidas realizadas por los sensores que componen la red. El funcionamiento consiste en que cada cierto período de tiempo se enviarán los datos de los sensores a dicha estación, la cual tiene todo el poder computacional. En la Figura 3 se puede ver un ejemplo de este tipo de redes:

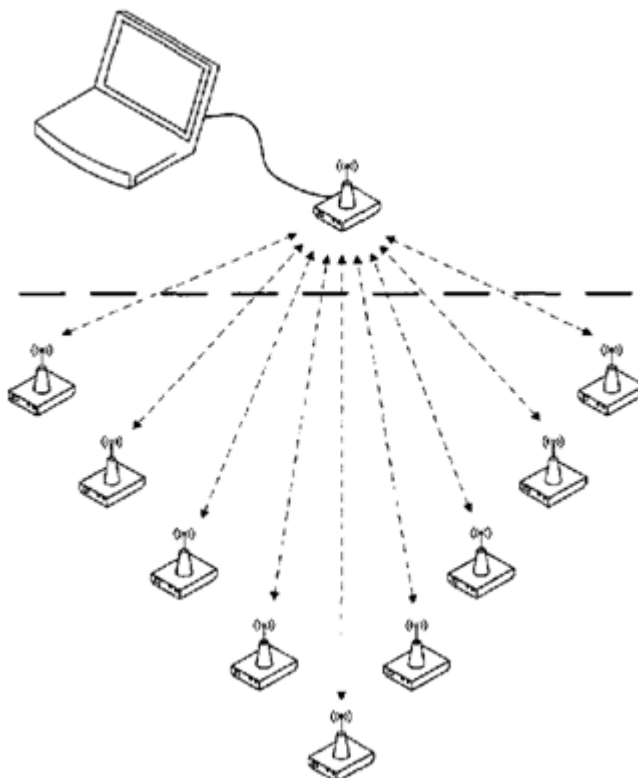


Figura 3 : Ejemplo de una red con procesamiento centralizado.

La principal ventaja de este tipo de redes reside en que no son necesarios nodos o sensores con grandes capacidades de cómputo, puesto que todas estas operaciones se llevan a cabo en la estación base. Este hecho hace que los componentes de la red sean sensores muy simples, y por consiguiente, baratos. Otra de las ventajas de este modelo de procesamiento es la estabilidad ante el agotamiento de baterías de alguno de los nodos de la red, puesto que no será necesario reconfigurarla y el sistema podrá seguir funcionando correctamente.

En el ámbito de las aplicaciones de localización, este tipo de procesamiento tiene una clara ventaja, debido al hecho de que posee un conocimiento bastante completo de la red y tiene acceso a un mayor número de medidas con lo que sería posible obtener estimaciones más precisas de la posición del blanco a localizar.

Por otro lado, el procesamiento centralizado tiene varios inconvenientes que no lo hacen adecuado para muchas aplicaciones, como por ejemplo:

2.1 PROCESAMIENTO EN REDES DE SENSORES

- Robustez: Como ya se ha comentado, este tipo de procesamiento se basa en la utilización de una estación central. Este hecho puede considerarse como una ventaja y un inconveniente al mismo tiempo, puesto que se está consiguiendo minimizar las tareas que tendrán que llevar a cabo el resto de sensores de la red, a cambio de que todo el funcionamiento de la red dependa de ese nodo central o estación base. De forma que si en algún momento la estación base quedara aislada (no fuera posible acceder a ella desde ningún otro nodo de la red), el procesamiento no podría llevarse a cabo, y la aplicación para la que se diseñó la red de sensores no funcionaría correctamente. Con lo que se puede considerar que este tipo de procesamiento es poco robusto ante cualquier tipo de problema o anomalía (ya sea de comunicación o computo) en la estación base.
- Elevado número de comunicaciones: Tal y como se había comentado, la estación base recibirá las medidas realizadas por cada uno de los sensores, con lo que la cantidad de tráfico que circula por la red en cada instante es muy elevada, ya que todos los sensores envían sus medidas, útiles o no, a la estación base. Por este motivo, el consumo energético de los sensores que componen la red se verá incrementado considerablemente. Este hecho se ve agudizado en los sensores cercanos a la estación base puesto que deben encaminar una mayor cantidad de paquetes hacia él, ya que la información de los sensores lejanos va pasándose de sensor a sensor porque si no se producirían muchas pérdidas. Tener tantas comunicaciones hace que se necesite un mayor ancho de banda, por lo que el coste de la red se verá incrementado.
- Escalabilidad: El aumento en el número de sensores que componen la red conlleva un aumento en el número de medidas realizadas, y en los paquetes que deberán enviarse a la estación base. Con lo que se puede decir que este método no es recomendable para redes con un número elevado de nodos. Además al aumentar el número de medidas aumenta la cantidad de información no relevante que se envía por la red, ya que sólo la información de los sensores que estén cercanos al objetivo será la que sirva para la estimación de la posición de dicho objetivo.
- Retardo: La estación base necesitará tener todos los datos de todos los sensores para completar su procesamiento. Esto hará que la estación tenga que esperar a tener todos esos datos para poder realizar la estimación. Debido a esto se producirá un retardo, puesto que la información de los sensores más lejanos es la que más tardará en llegar, puesto que tiene que pasar por varios sensores hasta llegar a la estación.

A pesar de estos inconvenientes el procesamiento centralizado resulta útil para modelos de red en los que los sensores tienen una capacidad de cómputo limitada y tienen acceso a un nodo central capaz de realizar todo el procesamiento.

Será necesario analizar estos y muchos otros factores para decidir qué modelo de procesamiento se adecúa mejor a las necesidades del problema o aplicación que se desee implementar. Así mismo deberá existir un compromiso entre el coste de la red y las prestaciones de la aplicación.

2.1.2 Procesamiento Distribuido

El procesamiento distribuido aparece como la solución a muchos de los problemas planteados por el procesamiento centralizado, explicado en el apartado anterior. En este caso cada nodo tiene la capacidad de cómputo suficiente como para recibir, almacenar y procesar datos. De esta forma cualquier nodo de la red podrá ser designado como nodo central o nodo líder, y será el encargado de recolectar y procesar las medidas realizadas por sus vecinos más próximos.

Normalmente las aplicaciones de las redes de sensores tratan de detectar eventos dinámicos, por lo que el sensor elegido como líder irá cambiando según se vea modificado el evento a detectar. Será necesario fijar un criterio para la selección del líder, algunos de los más utilizados son los criterios de información, que designarán como líder aquel sensor cuyas medidas aporten una mayor información sobre el evento, y el de potencia, que elegirá como líder aquel sensor cuya potencia recibida sea más elevada.

Este tipo de procesamiento también se presenta como un modelo de ahorro energético para los sensores, puesto que el nodo líder será el encargado de activar a sus nodos vecinos para que le envíen la información que tengan en cada instante, permitiendo así que el resto de la red permanezca inactiva.

La Figura 4 muestra un ejemplo de una red de sensores con procesamiento distribuido para una aplicación de localización (en este caso el elemento a localizar sería el coche que aparece en la imagen). Se puede apreciar cómo el nodo central o líder (representado como un punto negro en la figura) va variando a medida que el objetivo se va moviendo por todo el área de estudio.

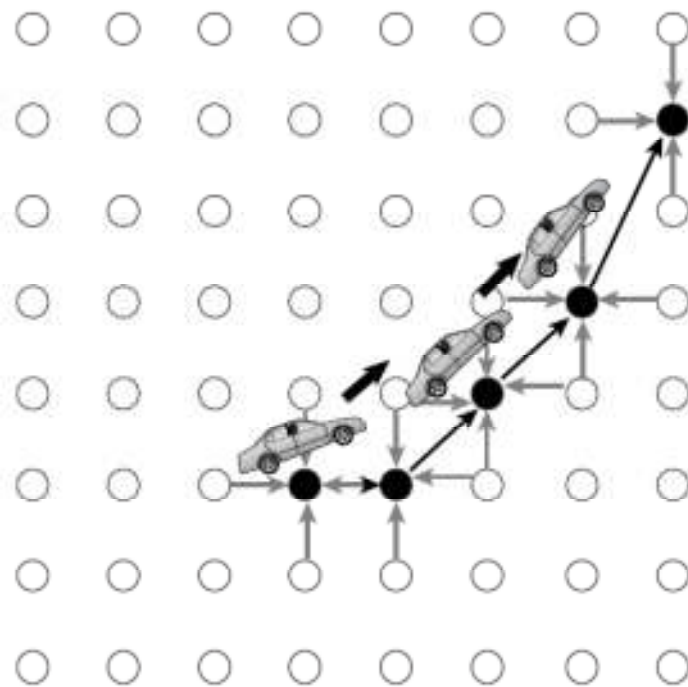


Figura 4 : Ejemplo de una red con procesamiento distribuido. [18]

2.1 PROCESAMIENTO EN REDES DE SENSORES

Este tipo de procesamiento se utiliza en numerosas aplicaciones debido a sus grandes ventajas, como por ejemplo:

- Robustez: Al contrario de lo que ocurriría con el procesamiento centralizado, en este caso el procesamiento no lo realiza un único nodo, si no que cualquier nodo de la red tiene una capacidad de cómputo suficiente como para asumir el papel de nodo central y completar las tareas de procesamiento necesarias para el correcto funcionamiento de la aplicación. De esta forma, se consigue una gran robustez para la aplicación y para la red en sí.
- Número de comunicaciones: El número de comunicaciones necesarias para este tipo de procesamiento es bastante reducido, en comparación con el modelo centralizado. Esto se debe a que sólo se realizarán comunicaciones entre el nodo líder y sus vecinos.
- Consumo energético: Como se había indicado, este procesamiento permite un gran ahorro energético puesto que en cada instante sólo permanecerán activos el nodo líder y sus vecinos más próximos, evitando así un consumo energético innecesario del resto de nodos de la red, por lo que se consigue un mayor aprovechamiento de las baterías.
- Escalabilidad: En el caso de que la red tenga que aumentar el número de sensores que la componen, no se produce ningún problema adicional puesto que ese aumento no implica que se produzca un aumento en el número de comunicaciones que deben llevarse a cabo dentro de la misma.
- Calidad de la información enviada: En este caso, cuando ya se ha elegido un líder, la calidad de la información que se envía por la red es toda útil. Esto se debe a que el líder estará cerca del objetivo, y como sólo los vecinos son los que envían la información al líder, estos nodos también estarán cerca del objetivo y por tanto sus medidas aportarán información a la estimación.

Pero este modo de funcionamiento, no tiene sólo ventajas en su aplicación, sino que también existen algunas desventajas. La primera de ellas es que el nodo líder no tiene una visión global de toda la red, por lo que a la hora de realizar la estimación puede tener más probabilidad de error, ya que sólo tiene en cuenta la información de sus vecinos. Pero si se tiene en cuenta que al ser el líder, es el que “teóricamente” está más cerca del objetivo, pues entonces con esa información tendría que ser suficiente para realizar la estimación de la trayectoria del objetivo a seguir, por lo que tampoco es tanto inconveniente.

El tamaño de la red influye mucho en la elección del modo de funcionamiento, en el caso de que la red sea pequeña, este modo de funcionamiento, no interesa, porque hay un procesamiento previo en la elección del líder, que será innecesario si el tamaño es pequeño y la cantidad de información que se necesite utilizando el otro modo no sea tan grande. Pero cuando la red ya empieza a tener un número de sensores considerable, este tipo de funcionamiento sí que interesa, puesto que se consigue reducir la cantidad de información procesada en la red, con ello el ancho de banda, y además se produce un ahorro de la energía consumida por las baterías de los nodos sensores, ya que llegados a un estado estable, con un líder bien elegido, los nodos que envían información son sólo los que están cercanos al objetivo, consiguiendo así una mayor eficiencia de la red.

2.2 Despliegue de la Red

El despliegue de los sensores que componen la red se llevará a cabo dentro de un determinado área. Se asumirá que las dimensiones de dicha área en los ejes x e y son dim_x y dim_y , respectivamente. Para las simulaciones que se realizarán en el Capítulo 5, se considerará que $dim_x=dim_y$, sin que esto suponga una pérdida de generalidad para los resultados obtenidos.

Dentro de dicha región se desplegarán N sensores, cuyas posiciones se denotarán como $[x_i, y_i]$, $i = 1, \dots, N$. La forma en la que se fijan estas posiciones marcará el tipo de despliegue que se ha realizado en esta red.

Básicamente existen dos tipos de despliegue: el aleatorio, en el que las posiciones de los sensores siguen una determinada distribución de probabilidad, y el determinista, en el que los sensores son posicionados manualmente en una determinada ubicación. Ambos modelos de despliegue resultan muy interesantes según las circunstancias en las que se quiere desarrollar esta aplicación (en lugares poco accesibles resultaría más sencillo realizar un despliegue aleatorio en lugar de determinista). Por esta razón, se ha optado por analizar la utilización de ambos tipos de despliegue y poder comparar así sus prestaciones a la hora de implementar aplicaciones de detección y localización de blancos.

2.2.1 Despliegue aleatorio de sensores

Tal y como ya se ha indicado, el despliegue aleatorio es aquel en el que las posiciones que ocupan los sensores vienen dadas por una determinada distribución de probabilidad (uniforme, gaussiana, etc.). Este tipo de despliegue es el más general puesto que puede ser utilizado en cualquier escenario por complejo e inaccesible que sea.

La distribución más utilizada para estos despliegues es la uniforme, según la cual todos los puntos del área de estudio tienen la misma probabilidad de ser ocupados por un sensor de la red. Teniendo en cuenta las dimensiones de nuestro escenario, la pdf (*probability density function*) de la posición de un sensor vendrá dada por:

$$p(x_i, y_i) = \frac{1}{dim_x} \cdot \frac{1}{dim_y} \quad (1)$$

Donde se cumple que $0 \leq x_i \leq dim_x$ y $0 \leq y_i \leq dim_y$.

En la Figura 5, se puede ver un ejemplo de este tipo de despliegue. Como se puede apreciar existen zonas descubiertas en las que no se ha posicionado ningún sensor, y otras en la que se tienen varios sensores muy próximos. Este hecho es uno de los problemas que plantea este tipo de despliegue puesto que al considerar que la posiciones de los sensores son independientes entre sí, no se tiene en cuenta si ya existía un sensor cercano a ese punto ni se da una mayor probabilidad de aparición a las zonas que han quedado descubiertas. Por esta razón todas las posiciones son igualmente posibles para un sensor cuando a éste se le asigna su posición.

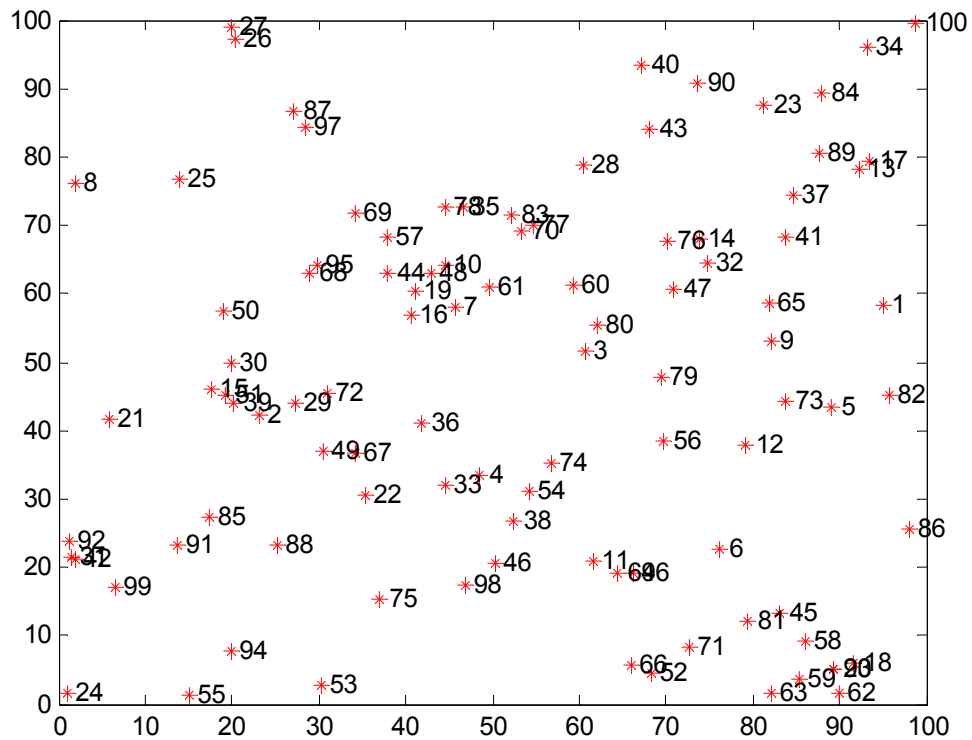


Figura 5: Ejemplo de la distribución de sensores siguiendo una variable aleatoria uniforme.

En esta figura se puede observar cómo se han posicionado los sensores en una ejecución de dicho tipo de despliegue. Los símbolos rojos que aparecen representados en la figura muestran la posición de cada uno de los sensores en la red. En el eje x y en el eje y vienen representadas las dimensiones del área de despliegue, que en este ejemplo es un cuadrado de 100 x 100.

2.2.2 Distribución de los sensores en forma de rejilla

Esta distribución de los sensores consiste en mantener una distancia constante entre los sensores tanto en un eje como en el otro. Con esto se consigue que los sensores formen una rejilla en todo el área de despliegue. De esta forma esta distribución se asegura de que todas las zonas están abarcadas por algún sensor, cerciorándose por tanto de que cualquier trayectoria que siga el objetivo pueda ser detectada, puesto que todos los rincones pueden ser medidos por algún sensor. Esta rejilla en un escenario real, no suele ser perfecta puesto que en la instalación de los sensores puede darse el caso en el que se produzcan variaciones, como que el área no sea totalmente un rectángulo, que el sensor se desplace en la instalación unos milímetros, etc. Por este motivo se introduce en el estudio una variación en las posiciones de la rejilla. Esta variación consiste en introducir a cada posición una variable aleatoria normal de media cero y varianza uno. Un ejemplo de la distribución de los sensores en forma de rejilla se puede observar en la Figura 6.

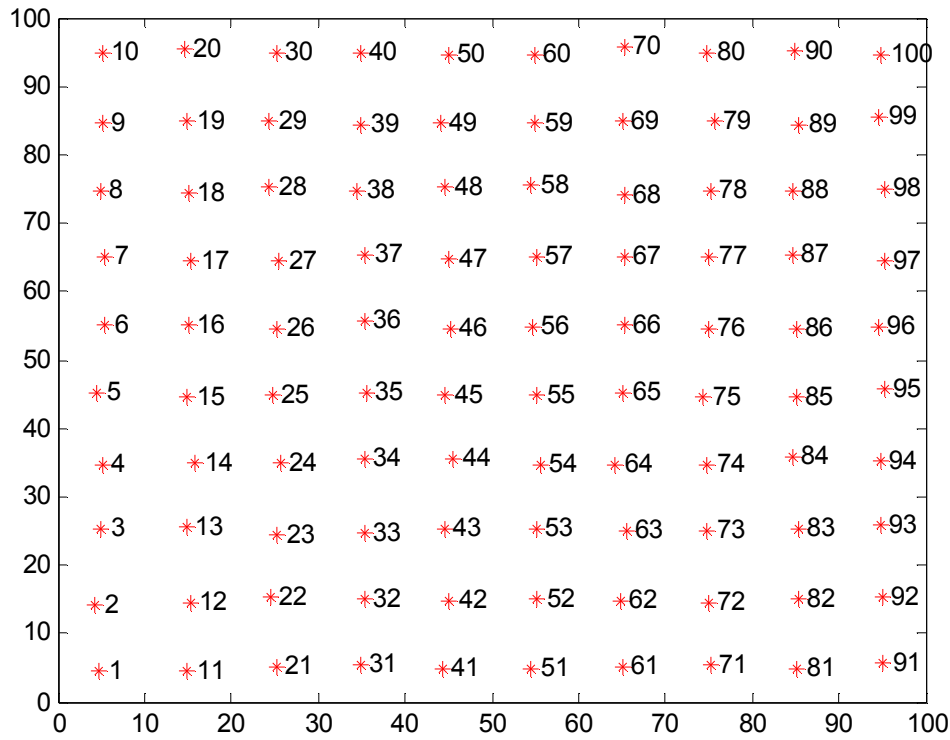


Figura 6: Distribución de los sensores en forma de rejilla cubriendo completamente el área de estudio.

En esta figura se puede observar cómo se han posicionado los sensores en una ejecución de dicho tipo de despliegue. Los símbolos rojos que aparecen representados en la figura muestran la posición de cada uno de los sensores en la red. En el eje x y en el eje y vienen representadas las dimensiones del área de despliegue, que en este ejemplo es un cuadrado de 100 x 100.

2.3 Localización de blancos

En el apartado 1.1.3 se mencionó la importancia que tenían las aplicaciones de localización y seguimiento en el ámbito de las redes de sensores. Generalmente estas aplicaciones plantean la localización como la solución a un sistema dinámico de la forma:

$$\mathbf{x}_t = f(\mathbf{x}_{t-1}, \mathbf{u}_t) \quad (2)$$

$$\mathbf{z}_t = g(\mathbf{x}_t, \mathbf{v}_t) \quad (3)$$

Donde $\mathbf{x}_t = [x_{1,t}, x_{2,t}, \dot{x}_{3,t}, \dot{x}_{4,t}] \in \mathbb{R}^4$ es un vector que representa el estado en el que se encuentra el objetivo, cuyos elementos son la posición y la velocidad del objetivo en el sistema de coordenadas cartesiano bidimensional en el instante t . \mathbf{u}_t es la componente de ruido y la función $f(\cdot)$ relaciona el vector de estado actual con el anterior y el ruido (es decir, la función de transición de estado). \mathbf{z}_t es la medida del sensor en el instante t , $g(\cdot)$ es la función de medida (se definirá en el apartado 2.4), y \mathbf{v}_t es la observación de ruido en cada medida tomada por el sensor. Se asume que \mathbf{u}_t y \mathbf{v}_t son estadísticamente independientes.

2.3.1 Modelo de movimiento del blanco

El modelo de movimiento que sigue el blanco que se quiere localizar vendrá determinado por la ecuación (4). Existen muchos modelos según la tipología del blanco a seguir: lineal, uniformemente acelerado, etc.

Para la realización de este proyecto se ha optado por uno de los modelos más utilizados en todas las aplicaciones de localización y que viene dada por [19]:

$$\mathbf{x}_t = \mathbf{G}_x \mathbf{x}_{t-1} + \mathbf{G}_u \mathbf{u}_t \quad (4)$$

Donde las matrices $\mathbf{G}_x, \mathbf{G}_u$ vienen definidas por:

$$\mathbf{G}_x = \begin{pmatrix} 1 & 0 & T_s & 0 \\ 0 & 1 & 0 & T_s \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (5)$$

$$\mathbf{G}_u = \begin{pmatrix} \frac{T_s^2}{2} & 0 \\ 0 & \frac{T_s^2}{2} \\ T_s & 0 \\ 0 & T_s \end{pmatrix} \quad (6)$$

Siendo T_s el período de muestreo, y \mathbf{u}_t es un vector 2×1 que representa un proceso de ruido Gaussiano con media cero y matriz de covarianza conocida $\mathbf{C}_u = \text{diag}\{\sigma_{u1}^2, \sigma_{u2}^2\}$.

La posición inicial del blanco, denotada como \mathbf{r}_0 , estará uniformemente distribuida dentro del escenario que se está utilizando, y definida como $\mathbf{r}_0 \sim U[0, \mathbf{dim}]$, $\mathbf{dim} = [\text{dim}_x, \text{dim}_y]$. De igual forma, la velocidad inicial del blanco se definirá como una distribución uniforme comprendida entre 0 y una determinada velocidad máxima, $\mathbf{r}_0 \sim U[0, \mathbf{V}_{max}]$, $\mathbf{V}_{max} = [V_{max}^x, V_{max}^y]$.

Además se va a controlar que la trayectoria quede siempre dentro del área de despliegue, de forma que si el objetivo está muy cerca de la frontera se cambiará el signo de su velocidad para que se vuelva hacia dentro del área, produciendo trayectorias con rebotes.

Un ejemplo de varias trayectorias generadas con este modelo se ven representadas en la Figura 7:

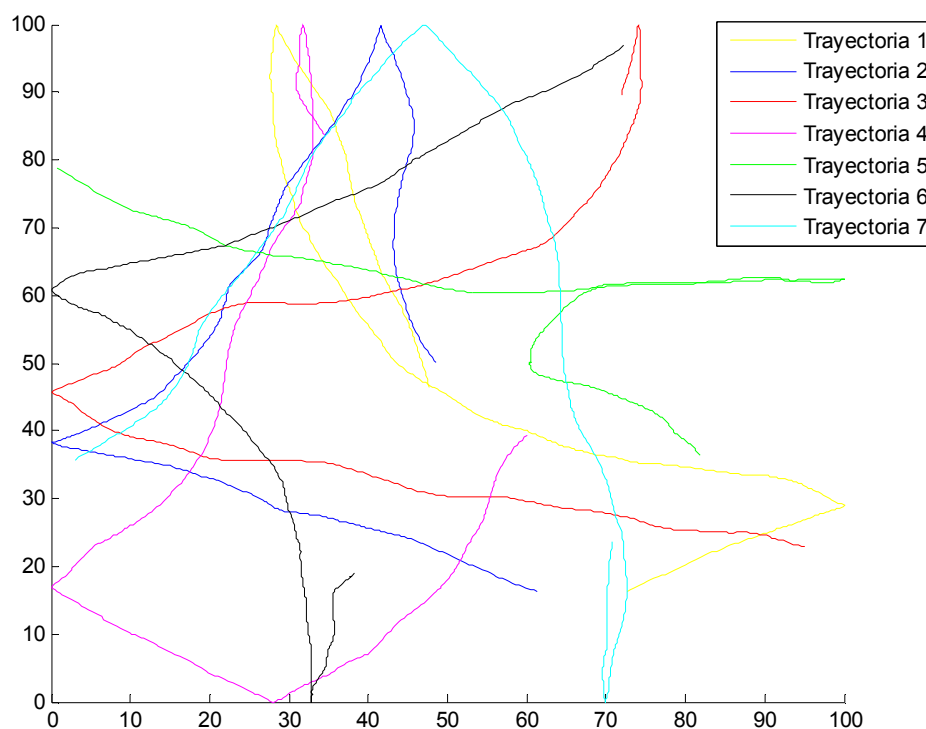


Figura 7 : Representación de siete trayectorias generadas con el modelo de movimiento.

2.4 Modelo de Medida

En el caso de seguimiento que es el que se va abordar en este proyecto, se describen dos tipos de sensores acústicos, el sensor acústico de amplitud y el sensor acústico DOA (Direction Of Arrival), que mide la dirección de llegada.

El sensor acústico de amplitud mide la amplitud del sonido que se está recibiendo en el micrófono, y estima la distancia hasta el blanco en movimiento mediante la ley de atenuación del sonido. En cambio, el sensor acústico DOA es un array de pequeños micrófonos. Este sensor puede determinar la dirección desde la cual procede el sonido utilizando las técnicas de conformación de haz. En concreto, la estimación se basa en la fuerza con la que se recibe la señal y en la diferencia de tiempo de llegada (TDOA).

En general, se habla de sensores que detectan potencia, independientemente de la señal que se esté midiendo. Este último caso es el que se ha elegido para realizar este estudio puesto que es un caso más general. Dentro de los sensores que miden potencia

existen distintos tipos, la diferencia es el modelo de medida que utiliza cada uno. Dentro de estos modelos de potencia se ha elegido el que se ha creído más conveniente para el estudio que se está realizando.

Se va a explicar ahora el modelo de medida que utiliza el sensor que mide potencia que se ha elegido. El valor RMS (Root Mean Square) de la amplitud de la medida z (un escalar en este caso) con la distancia entre el sensor y el objetivo, y un valor inicial de potencia, tal y como se especifica en la ecuación (7).

$$z = \frac{P_0}{(1 + \alpha)d^2} + w \quad (7)$$

donde P_0 es un valor de potencia inicial recibida, d es la distancia que hay entre el sensor y el objetivo que se está midiendo, y α es el factor que va a controlar la caída de la potencia recibida en función de la distancia. El parámetro w representa el ruido que hay en el entorno, que en este caso, será modelado como un ruido blanco gaussiano de media cero y varianza σ^2 .

En Figura 8 se puede observar cómo varía la potencia que obtiene el sensor en función de la distancia que hay entre dicho sensor y el objetivo. De esta figura se puede observar que a partir de una cierta distancia, la potencia recibida será nula. El valor de este límite para este caso, se va a considerar a partir de una distancia de 15.

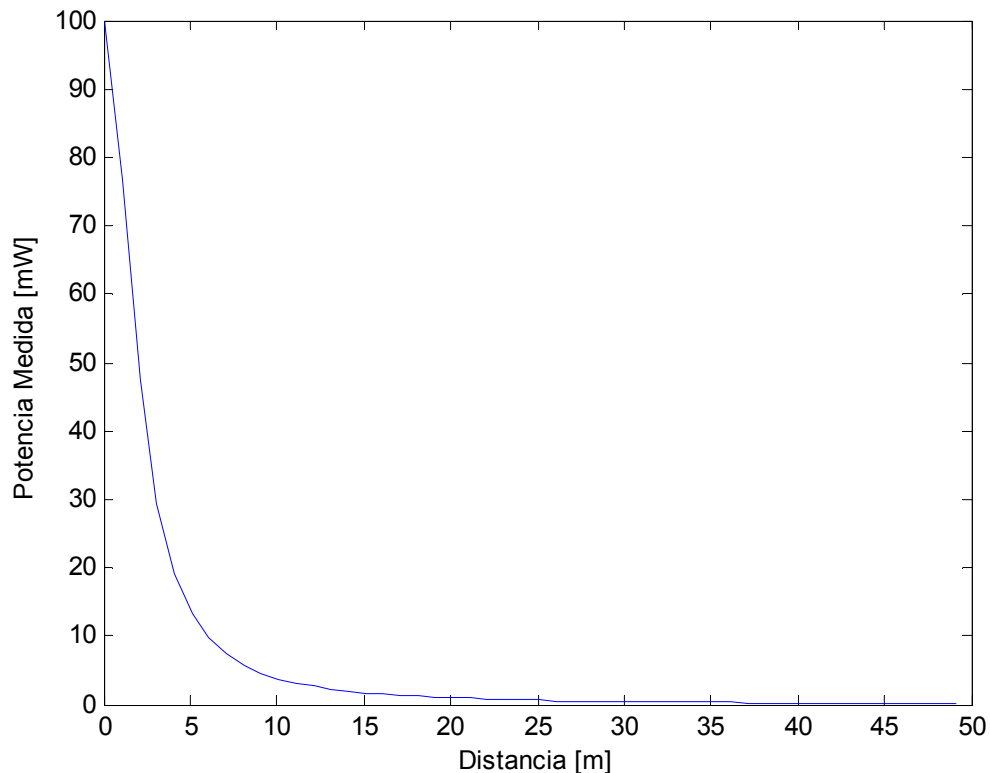


Figura 8 : Potencia recibida por el sensor en función de la distancia ($\alpha = 20$ dB y $\alpha = 0.25$).

Capítulo 3

Detección en Redes de Sensores

A lo largo de este capítulo se va a desarrollar todo lo que tiene que ver con la detección en este tipo de redes. En concreto se hablará de la detección en general, y después se concretará la detección para las redes de sensores. En el último punto se explicarán y desarrollarán las distintas técnicas que han sido estudiadas en este proyecto para entender cuál es la forma de aplicación de cada una de ellas. Éstas serán las que en los resultados experimentales se comparen para obtener las conclusiones pertinentes.

3.1 Detección de señales

La teoría de la detección de señales sirve para determinar la habilidad que existe para separar la señal del ruido que existe en el entorno, y que se añade a las medidas realizadas. Cuanto mayor sea la habilidad de separar la señal real del ruido, mejores resultados se obtendrán en la detección de la señal que se quiere calcular. De acuerdo a esta teoría, existen distintas técnicas que determinan cómo un sistema de detección detectará la señal, lo que hacen es calcular unos umbrales que determinarán a partir de dónde se considerará señal y de dónde ruido. La teoría explica cómo se fijan estos umbrales, e incluso cómo van evolucionando en el caso de que el umbral no sea siempre el mismo, para detectar la señal en movimiento. La elección de estas técnicas dependerá de la señal a seguir y de cómo sea el escenario de trabajo.

CAPÍTULO 3 DETECCIÓN EN REDES DE SENSORES

La teoría de la detección tiene aplicaciones en muchos campos como en diagnósticos de cualquier tipo, control de calidad, telecomunicaciones, y psicología. El concepto es similar al de una señal con un ruido usada en ciencias y las matrices confusas usadas en inteligencia artificial. También es útil en el mantenimiento de alarma, donde es importante separar los eventos importantes del ruido de fondo.

La teoría de la detección de señales ha evolucionado a partir del desarrollo de los equipos de comunicaciones y radar en la primera mitad del siglo. Estos conocimientos emigraron a la psicología, inicialmente como parte de la percepción y sensación, y en los años 50 y 60 como un intento de entender algunas de las características del comportamiento humano cuando se detectan estímulos muy débiles que no se han explicado por las teorías tradicionales de los umbrales.

La teoría tradicional de umbrales lo que tiene de diferente es que es el sujeto el que toma una decisión, un acto cognitivo, en cuanto a si la señal está presente o no. El acto de determinar si se produjo un estímulo o no, tiene ahora un componente cognitivo. Lo que ocurre en esta situación, es que si la señal está presente, la persona puede decidir si la señal está o no está. En el caso de que decida que la señal está, entonces la decisión será acertada, si por el contrario decide que no está entonces se habrá equivocado. Si la señal no está, el sujeto podrá volver a decidir si la señal está o no está. En el caso de que decida que sí que está, se producirá una falsa alarma, y si por el contrario decide que no está, se habrá producido un rechazo correcto. Estas son las posibilidades que se tienen como resultados de la detección de señales. Esto se puede ver representado en la Tabla 1.

Tabla 1 : Representación de los posibles estados de las decisiones tomadas.

		Decisión tomada	
		Señal	Ruido
Medida realizada	Señal	Acierto	Fallo
	Ruido	False Alarma	Rechazo correcto

3.1.1 Formulación

La detección de señales se basa en determinar si las probabilidades de verosimilitud cumplen $p(H_1|x) > p(H_0|x)$, donde H_1 es la hipótesis de que lo que se ha medido sea señal y H_0 la hipótesis de que lo que se ha medido sea sólo ruido. La variable x representa la medida que se ha tomado de la señal. Por lo que se buscará el umbral que esté entre el límite de esta suposición y la contraria.

3.2 Detección en redes de sensores

Las redes de sensores pueden estar situadas en muchos lugares y con fines muy distintos unas de otras. Pueden estar en un supermercado para determinar los lugares más visitados por los consumidores, pueden estar integradas en un sistema de seguridad para determinar el recorrido que hacen los visitantes al lugar donde se encuentra el sistema, pueden estar situados en determinadas zonas de la naturaleza para averiguar cómo se mueven ciertos animales, en una fábrica para detectar fallos en el material, o en un hospital para llevar un seguimiento de las constantes vitales, por ejemplo. Todas estas redes se basan en detectar algo. Esta detección es la base del sistema, puesto que el resultado depende de lo que se esté detectando. Por este motivo, es muy importante elegir bien el algoritmo que va a ser capaz de distinguir entre la señal real y el ruido del sistema.

La detección es importante puesto que puede ayudar al ser humano a ver cosas que a simple vista, o incluso, observando muy detenidamente, no podría percibir. También puede ayudar a ver que ciertos factores están relacionados incluso cuando al cerebro no se le habría ocurrido esa conexión simplemente observando. De la detección además se puede aprender cómo se comportan determinados seres vivos, materiales, máquinas, etc., de una forma más sencilla. La detección en este tipo de redes ayuda a que el ser humano tenga un conocimiento mayor y más exhaustivo de todo lo que le rodea, además puede ayudarle en su vida, ya que si por ejemplo, en una fábrica, se detectan mejor los problemas que haya en el material que se está fabricando, entonces los objetos que se hagan con ese material serán de mayor calidad y por lo tanto tendrá una vida mayor, por lo que será bueno para el ser humano.

En el caso que se está estudiando, la detección consiste en determinar la trayectoria que sigue un objetivo. La importancia de detectar una trayectoria puede ser muy grande, dependiendo del objetivo para el que se use. Por ejemplo en una tienda, como ya se ha mencionado anteriormente, se puede estudiar la trayectoria que siguen los consumidores, y se puede ver cuáles son los pasillos más visitados, cuáles son dónde están más tiempo, etc. Esto puede servir, para hacer una reubicación de los productos y así poner los objetos con los que se saca mayor beneficio en los pasillos más visitados, esto produciría un aumento en el capital de la tienda. Otro ejemplo, podría ser integrar los sensores en un sistema de seguridad, para poder determinar el recorrido que hacen los ladrones o los posibles ladrones, y así poner algún tipo de mecanismo que impida que los futuros ladrones se cuelen por el mismo sitio y con la misma facilidad. Hay otros muchos ejemplos que se podrían mencionar, pero lo importante es destacar que con la detección de una simple trayectoria se puede obtener mucha información que a simple vista no se puede observar. Si a esa detección de la trayectoria se le pueden añadir otros factores de tiempo, espacio, lugar y movimiento, entonces la información que se obtiene es mucho más productiva, y se pueden llegar a obtener determinados comportamientos que están relacionados, y que pueden ayudar a que en el futuro no se cumplan, o se utilicen para obtener algún tipo de beneficio, por ejemplo.

Por lo tanto, se podría decir que la detección consiste simplemente en determinar un umbral, que puede ir variando a lo largo del tiempo o no, a partir del cual se decide si hay señal real o simplemente es ruido, y por dicha razón el objetivo no está en el área de despliegue, es decir, sólo se quiere detectar si el objetivo está o no está.

CAPÍTULO 3 DETECCIÓN EN REDES DE SENSORES

La hipótesis nula, es decir H_0 , consiste en afirmar que lo que se está detectando es simplemente ruido, y la hipótesis alternativa H_1 , por el contrario, es la que afirma que el objetivo está siendo detectado por el sistema. Esto se puede ver expresado en la ecuación (8).

$$\begin{cases} H_0: & z < \text{umbral} \\ H_1: & z \geq \text{umbral} \end{cases} \quad (8)$$

Donde z es la medida que cada nodo sensor toma cada cierto periodo de tiempo de lo que hay a su alrededor, y el umbral es un valor que según el algoritmo utilizado se decidirá de una forma u otra y será el que determine en qué hipótesis nos encontramos.

En la Figura 9 se puede observar lo que ocurre con la medida tomada. En ella se representa la función de densidad de la probabilidad del ruido que hay en el ambiente, y además la función de densidad de probabilidad de la señal que toma el sensor, es decir, de la señal real más el ruido. En función de dónde se coloque el umbral, se van a definir las áreas de la probabilidad de falsa alarma y de detección. En la figura se puede ver que el umbral está representado con una línea negra, y el área sombreada es la probabilidad de falsa alarma. En función del valor que se elija el umbral se moverá a la izquierda, si hay mayor probabilidad de falsa alarma, y a la derecha si hay menor.

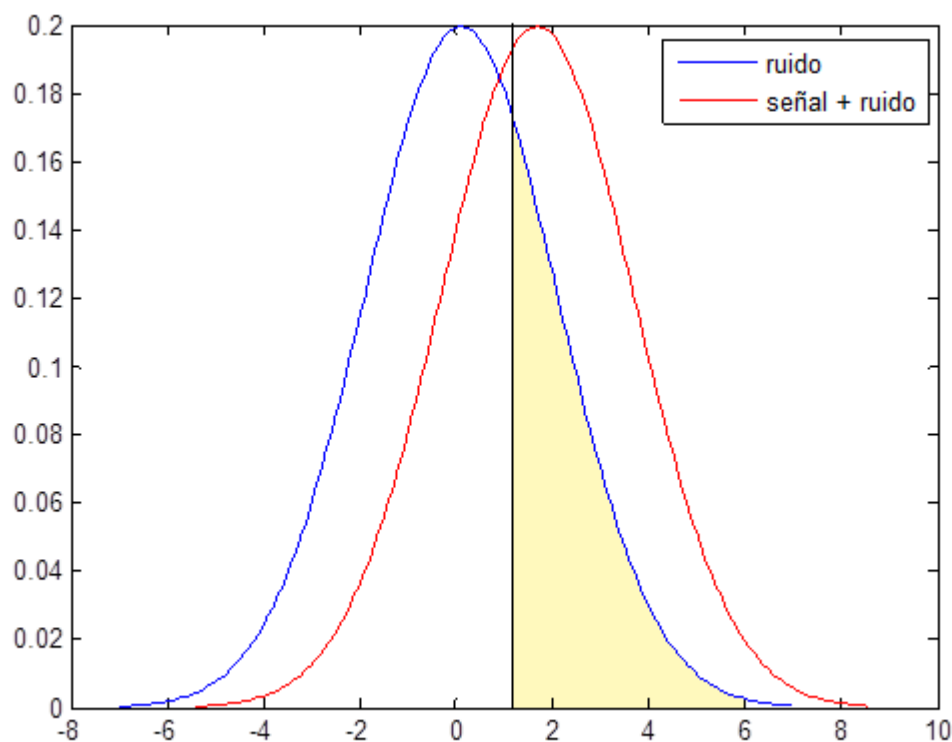


Figura 9 : Representación de la función densidad de probabilidad.

3.3 Técnicas de Detección

Las técnicas de detección que se analizarán en este proyecto son dos de las más utilizadas en los distintos ámbitos en los que se aplica la detección (rádar, etc.), es decir, la técnica CFAR (Constant False Alarm Rate Detector) y la técnica Decision Fusion Rule (DFR).

La técnica CFAR permite calcular un umbral de decisión a partir de las medidas que se van obteniendo en cada instante. De esta forma se consigue que cada sensor sea capaz de ajustar el umbral de detección al ambiente en el que se encuentra. Se trata de una técnica de detección muy adecuada para su utilización en redes con procesamiento centralizado, puesto que así cada sensor será capaz de fijar su umbral de acuerdo a la información que tiene del entorno y evitar el envío de información innecesaria a la estación base.

Por otro lado, se tiene la técnica de Decisión Fusion Rule que permite la utilización de un nodo central o líder para la toma de la decisión. Tiene un funcionamiento muy similar al que se plantea en el procesamiento distribuido: el nodo líder recopila la información de sus vecinos, que en este caso serán las decisiones individuales que cada uno de ellos haya tomado, y a partir de ésta decidirá si se ha detectado algo o no. Debido a estas similitudes con el procesamiento distribuido, esta técnica de detección resulta la más idónea para este tipo de redes. Como se verá a continuación existen dos variantes de esta técnica, una que nos proporciona el umbral óptimo, y otra que da una simplificación del mismo.

3.3.1 Constant False Alarm Rate Detector

La detección Constant False Alarm Rate se refiere a una forma común de un algoritmo adaptativo que se usa en sistemas radar para detectar el objetivo sobre un fondo de ruido, desorden e interferencia.

En radar se basa en la recepción de los ecos de la señal emitida. Típicamente se reciben a través de una antena, se amplifican, se pasan por un convertidor y después por un detector que obtiene la envolvente de la señal (conocida como señal de vídeo). Esta señal de vídeo es proporcional a la potencia del eco recibido y se compone de la señal de eco que se quiere detectar y una potencia no deseable de ruido interno recibido y desorden e interferencia externas.

El papel del circuito de CFAR es determinar el umbral de potencia sobre el cual cualquier eco recibido se considerará originado por el objetivo. Si este umbral es demasiado bajo, entonces se detectarán más objetivos a expensas de incrementar el número de falsas alarmas. De forma contraria, si el umbral es demasiado alto, entonces se detectarán pocos objetivos, pero el número de falsas alarmas también será bajo. En muchos detectores radar, el umbral se establece con el fin de lograr una probabilidad de falsa alarma (o su equivalente, la tasa de falsa alarma o el tiempo entre falsas alarmas).

Si el escenario en el que los objetivos son detectados es constante en tiempo y en espacio, entonces el umbral fijado se puede elegir para que proporcione una probabilidad de falsa alarma determinada, regido por la función de densidad de probabilidad del ruido, que normalmente se asume gaussiano. La probabilidad de detección es por tanto una función de la SNR (signal to noise ratio, relación señal a ruido) del objetivo. Sin embargo, en muchos de los sistemas desplegados, las fuentes de desorden e interferencia no deseadas hacen que el nivel de ruido cambie tanto espacialmente como temporalmente. En este caso, se puede utilizar un umbral variante, donde el umbral vaya subiendo y/o bajando para mantener una probabilidad de falsa alarma constante. Esta es la detección conocida como CFAR.

Dicha técnica se puede utilizar de forma análoga para la detección en redes de sensores. A diferencia de las aplicaciones radar, los nodos de la red realizarán la detección sobre las medidas llevadas a cabo por cada uno de ellos.

A continuación se van a describir las distintas fases de la detección CFAR, en concreto tiene tres. La primera de ellas es la fase de inicialización en la que el umbral se inicializa en función del ruido que haya en el sistema, es decir, se supone que el ruido es gaussiano con una media y varianza determinadas, a partir de estos dos valores se hará la inicialización. La segunda fase consistirá en tomar las medidas por parte de cada sensor, esto está explicado en el apartado 2.4 dónde se explica el modelo de medida de los sensores. Pero esta fase se incluye en este algoritmo, puesto que las medidas tomadas por el sensor se utilizarán para compararlas con el umbral y por tanto determinar si se está detectando o no. Por último, la fase final consiste en actualizar dicho umbral para que en el instante siguiente se haya adaptado a los cambios. Después de este paso se irán repitiendo la segunda fase y la última simultáneamente para realizar las distintas detecciones.

La fase de inicialización consiste en primer lugar en inicializar la media $\mu_k(0)$ y la desviación típica $\sigma_k(0)$ con las características del ruido que se tendrá en el sistema. En este caso el ruido que se va a utilizar es un ruido Gaussiano. Para tomar estas medidas se deja al sistema funcionar sin que haya ningún objetivo dentro de la región de los sensores, y así tener una medida más realista del ruido que existe en el sistema. Esta primera fase es la fase de calibración. Una vez se han tomado estos datos iniciales, el sistema empieza a funcionar de forma recursiva. En primer lugar, cada nodo sensor k toma sus medidas de la potencia que está detectando en ese instante, a la que se va a denotar como $z_k(n)$. Esta medida tomada se compara con un umbral $T_k(n)$, y en función de si la medida es mayor o menor al umbral se tomará la decisión de si lo que se mide es algo válido o es ruido. El umbral se define como $T_k(n) = \mu_k(n) + C\sigma_k(n)$, donde C es una constante elegida para conseguir un valor de probabilidad de falsa alarma deseable de la siguiente forma:

$$P_{fa_i} = \frac{1}{\sqrt{2\pi}} \int_c^{\infty} \exp\left(-\frac{1}{2}u^2\right) du \quad (9)$$

La decisión $\gamma(n)$ se corresponde con lo que va a decidir cada sensor en cada instante de tiempo en el que se toma la medida al comparar con el umbral calculado.

$$\gamma(n) = \begin{cases} 1 & z_k(n) > T_k(n) \\ 0 & z_k(n) < T_k(n) \end{cases} \quad (10)$$

donde $\gamma(n) = 1$ indica que existe algún objetivo y 0 que lo que se está midiendo es simplemente ruido. Si $\gamma(n) = 1$, el umbral se va a mantener al mismo valor, para que éste no se dispare cuando el objetivo se acerque demasiado al nodo sensor, es decir, $T_k(n) = T_k(n-1)$, en otro caso, el umbral se actualizará de la forma:

$$\mu_k(n) = \alpha\mu_k(n-1) + (1-\alpha)z_k(n) \quad (11)$$

$$\sigma_k^2(n) = \alpha\sigma_k^2(n-1) + (1-\alpha)[z_k(n) - \mu_k(n)]^2 \quad (12)$$

$$T_k(n) = \mu_k(n) + C\sigma_k(n) \quad (13)$$

donde α es un factor de memoria cuyo valor está entre 0 y 1. Este parámetro permitirá indicar la importancia que se le da a los umbrales anteriores. En el capítulo 5 se hará un estudio más exhaustivo sobre el valor que debe tomar dicho parámetro.

3.3.2 Decision Fusion Rule

Este método actúa de forma distinta al anterior puesto que existe una cooperación entre los nodos sensores para realizar la detección. Como se mencionó anteriormente este algoritmo se basa en la utilización de un nodo sensor líder que se encargará de recopilar la información de sus vecinos y procesarla para determinar si existe un objetivo, o si por el contrario sólo se mide ruido.

Se asumirá que cada sensor i tiene una determinada probabilidad de falsa alarma y de detección, que vendrá dada por:

$$P_{fa_i} = \frac{1}{\sqrt{2\pi}} \int_C^\infty \exp\left(-\frac{1}{2}u^2\right) du = Q(C) \quad (14)$$

$$P_{di} = \frac{1}{\sqrt{2\pi}} \int_C^\infty \exp\left(-\frac{1}{2}(C - z_i)^2\right) dt = Q(C - z_i) \quad (15)$$

Donde la variable C representa un umbral elegido igual para todos los sensores, que se toma como referencia, y el valor de z_i representa la medida realizada por el sensor i -ésimo (explicado en el apartado 2.4).

De forma, que cada sensor tomará una decisión individual denominada $I_i = \{0,1\}$ ($i = 1, \dots, N$). Donde I_i toma el valor 1 cuando hay una detección válida y toma el valor 0 en caso contrario.

El siguiente paso en su implementación será la transmisión de la información recopilada por cada sensor al nodo líder para que éste calcule el umbral de decisión, Λ_0 , que determinará la decisión general del sistema. Existen dos formas de calcular este umbral, una es la solución óptima para la cual serán necesario conocer la decisión que ha tomado cada sensor y la medida que ha tomado en ese instante; y una aproximación al umbral óptimo para aquellos casos en los que resulta muy costoso tener acceso a toda esta información y los sensores se limitan a enviar la decisión que ha tomado cada uno de ellos. A continuación se verá el desarrollo de ambas opciones.

3.3.2.1 Optimal Decision Fusion Rule

Como se había comentado anteriormente, en este caso se calcula el umbral óptimo, puesto que se asume que el nodo líder tiene acceso a la información completa: medida realizada y decisión tomada por cada sensor vecino.

El valor obtenido Λ_0 se utiliza para determinar si lo que se ha medido es ruido o no. Esta regla viene detallada en [3].

$$\begin{aligned}\Lambda_0 &= \sum_{i=1}^N \left[I_i \log \frac{p_{d_i}}{p_{fa_i}} + (1 - I_i) \log \frac{1 - p_{d_i}}{1 - p_{fa_i}} \right] \\ &= \sum_{i=1}^N I_i \log \frac{p_{d_i}(1 - p_{fa_i})}{p_{fa_i}(1 - p_{d_i})} + \sum_{i=1}^N \log \frac{1 - p_{d_i}}{1 - p_{fa_i}}\end{aligned}\quad (16)$$

Por otro lado el valor de p_{fa_i} y p_{d_i} , viene definido en las ecuaciones (14) y (15). Se verá que la p_{fa_i} tiene el mismo valor para todos los sensores de la red puesto que será configurada para ello. En cambio la p_{d_i} depende de la medida que haya realizado cada sensor y debido a ello es necesario el envío de esta información al nodo líder.

El líder una vez que ha calculado el valor de Λ_0 , lo comparará con un umbral, que se decidirá en el momento de la implementación. Este valor se podrá elegir en función de los valores que tome Λ_0 en el sistema, y por lo tanto se podrá escoger el valor medio para que cuando esta variable supere el valor decidido, entonces se considere que se está detectando, y cuando ocurra lo contrario, se determinará que sólo es ruido.

3.3.2.2 Aproximación de Optimal Decision Fusion Rule

Este método en realidad es una aproximación del que se acaba de explicar en el apartado 3.3.2.1.

Anteriormente se vió que para el cálculo de Λ_0 era necesario conocer tanto la decisión que había tomado cada sensor como la medida que había obtenido cada uno de ellos. En muchos casos el acceso a tanta información no es fácil, por lo que se optó por simplificar el cálculo de Λ_0 para que sólo fuera necesario tener acceso a la decisión individual de cada nodo vecino. De esta forma el cálculo del umbral se haría del siguiente modo, tal y como se define en la ecuación (17).

$$\Lambda = \sum_{i=1}^N I_i \geq T \quad (17)$$

Es decir, el umbral Λ se calcula como la suma de las decisiones tomadas por el nodo líder y todos sus vecinos, simplificando así los cálculos a realizar y las comunicaciones necesarias (solo sería necesario enviar un bit de información).

Al igual que ocurría con la decisión óptima, será necesario comparar Λ con un umbral T , como se indica en la ecuación (17). La elección de este parámetro se detalla en el Capítulo 5.

Capítulo 4

Algoritmo de Localización

En este capítulo se va a explicar el algoritmo que se ha utilizado para estimar la trayectoria. Se va a explicar cómo se aplica dicho algoritmo al caso que se está estudiando a lo largo de este proyecto. En concreto se verá cómo va evolucionando el algoritmo en cada instante para conseguir estimar la trayectoria que sigue el objetivo, y a la vez conseguir minimizar el error.

4.1 Filtros de partículas

El filtro de partículas se va a encargar de obtener una buena estimación de la localización del sensor $x_{(t)}$ desde la historia de medidas $\overline{z_{(t)}}$. Lo que se quiere es que la estimación $\hat{x}(\overline{z_{(t)}})$ esté, en media, lo más cerca posible del valor real de $x_{(t)}$ de acuerdo a alguna medida. Es decir, la estimación debería minimizar el coste medio

$$\mathcal{E} = E[d(\hat{x}(\overline{z_{(t)}}), x_{(t)})] \quad (18)$$

donde $d(\cdot, \cdot)$ es una función de pérdidas para medir el rendimiento del estimador. Por ejemplo, $d(\hat{x}, x) = \|\hat{x} - x\|^2$ mide el cuadrado de la distancia entre el valor real y el estimado, es decir, es el caso que se va a utilizar para el estudio que se está realizando. Para esta función de pérdidas, el estimador es:

$$\hat{x}_{MMSE}^{(t)} = E [x^{(t)} | \bar{z}^{(t)}] = \int x^{(t)} p(x^{(t)} | \bar{z}^{(t)}) dx^{(t)} \quad (19)$$

El estimador es conocido como el estimador de mínimo error cuadrático medio (MMSE) [4]. La clave está en calcular la probabilidad $p(x^{(t)} | \bar{z}^{(t)})$ de la forma más eficiente y sencilla en la que se obtengan resultados aceptables.

El filtro de partículas se va a encargar de estimar la trayectoria que sigue el objetivo que se está detectando. Este método es una técnica sofisticada donde la estimación está basada en la simulación, también es conocido como el método secuencial de Monte Carlo (SMC). Principalmente se trata de una aproximación numérica muy extendida para solucionar distintos problemas estadísticos. Lo que se hace es, en cada iteración, se generan muestras de Monte Carlo de una variable de estado, con las que se realiza la aproximación a la función de densidad de probabilidad, este proceso se va repitiendo de forma recursiva.

Los filtros de partículas suelen utilizarse en la estimación de modelos Bayesianos, y son similares a los modelos conocidos como Markov Chain Monte Carlo (MCMC), destacando que los filtros de partículas diseñados correctamente pueden ser mucho más rápidos que los modelos MCMC. Además representan también una alternativa al filtro de Kalman ampliado (EKF) o al filtro de Kalman imperfumado (UKF), teniendo la ventaja de que si se tienen muestras suficientes se aproximan a la estimación Bayesiana óptima, obteniendo mejores resultados que con ambos filtros de Kalman nombrados.

En este caso, se van a utilizar los métodos SMC Bayesianos para resolver el problema de seguimiento de objetivos. Para determinar la localización y la velocidad a partir de una observación con ruido, se va a emplear un modelo de sensor no lineal, y se construirá un estimador SMC, basado en técnicas de muestreo secuencial por importancia. Este desarrollo permitirá calcular la función de densidad de probabilidad a posteriori del objetivo que se quiere localizar, basándose únicamente en las medidas obtenidas por los distintos sensores de la red.

4.1.1 Formulación del filtro de partículas

Se llama X_t a la variable de estado del objetivo a seguir en el instante de tiempo t , y se llama z_t a la medida realizada en el mismo instante t . Con esto se puede definir la ecuación dinámica de la variable de estado y de la medida realizada como:

$$\begin{aligned} X_{(t+1)} &= f(X_t) + u_t \\ z_t &= g(X_t) + v_t \end{aligned} \quad (20)$$

Donde las funciones $f(X_t)$ y $g(X_t)$ relacionan el estado anterior con el actual, y el estado actual con la observación realizada respectivamente. Estas funciones pueden ser lineales o no lineales, pero en este caso se va a estudiar el caso en el que $g(X_t)$ es una función no lineal. Por otro lado, las variables u_t y v_t se van a modelar como variables aleatorias Gaussianas de media cero y varianza Q y R , respectivamente, es decir, $u_t \cong N(0, Q)$ y $v_t \cong N(0, Q)$.

En el caso Bayesiano, el problema está en calcular la función de densidad de probabilidad a posteriori, $p(X_t|\bar{z}_t)$. A partir de la ecuación (20), se puede definir dicha función como:

$$p(X_t|\bar{z}_{t-1}) = \int p(X_t|X_{t-1}, \bar{z}_{t-1}) p(X_{t-1}|\bar{z}_{t-1}) dX_{t-1} \quad (21)$$

Suponiendo que se tiene un conjunto de muestras de la distribución, también llamadas partículas, y sus pesos correspondientes, $\{X_{t-1|t-1}^{(i)}, w_{t-1}^{(i)}\}$. Estas muestras forman un conjunto de medidas aleatorias que describen la función de densidad de probabilidad a posteriori en el instante t-1, como se puede observar en la Figura 10, donde las partículas están representadas con círculos amarillos y los pesos de las mismas, están representados con círculos rosas. En esta figura se puede observar cómo el volumen de las bolas que representan los pesos es mayor cuando la probabilidad de que el objetivo esté en esa partícula es mayor, y es menor cuando la probabilidad de que el objetivo esté en esa partícula es menor.

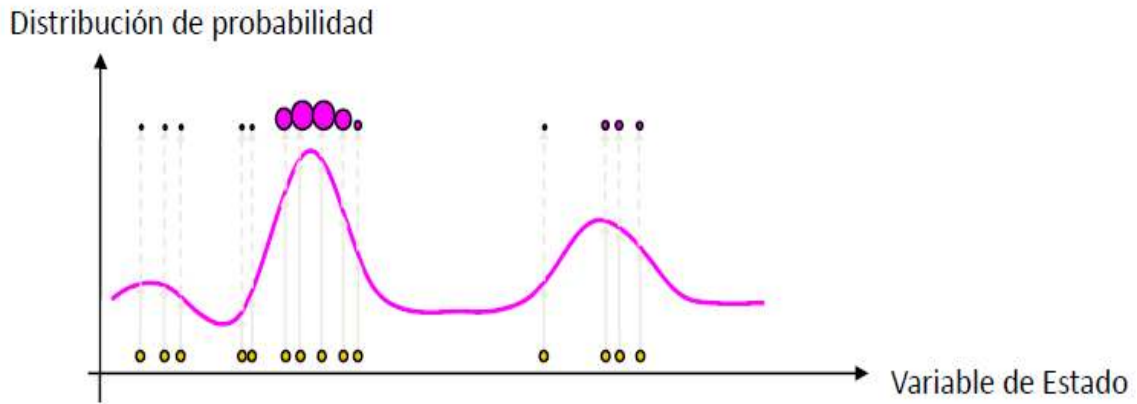


Figura 10: Ejemplo de partículas y coeficientes para la aproximación de una distribución de probabilidad.

De esta forma se puede escribir la función densidad de probabilidad a posteriori en el instante t-1 como:

$$p(X_{t-1}|\bar{z}_{t-1}) \approx \sum_{i=1}^N w_i^{(t-1)} \delta(X_{t-1} - X_{t-1|t-1}^{(i)}) \quad (22)$$

Si por otro lado, las partículas $X_{t-1|t-1}^{(i)}$ se dibujan a partir de una función de densidad de importancia, $q(X_{t-1}|\bar{z}_{t-1})$, se pueden definir los coeficientes que aparecen en la ecuación (22) como:

$$w_{t-1}^{(i)} = \frac{p(X_{t-1|t-1}^{(i)})}{q(X_{t-1|t-1}^{(i)})} \quad (23)$$

Para el caso secuencial, [22], denominado como Sequential Importance Sampling (SIS), en cada iteración se va a obtener la medida aleatoria $\{X_{t-1|t-1}^{(i)}, w_{t-1}^{(i)}\}_{i=1}^N$, que constituye una aproximación de $p(X_{t-1}|\bar{z}_{t-1})$, y se busca la aproximación de la función $p(X_t|\bar{z}_t)$ con un nuevo conjunto de muestras y de coeficientes. Usando las ecuaciones anteriores, se puede definir la ecuación de actualización de los coeficientes para cada partícula como:

$$w_t^{(i)} \propto \frac{p(z_t|X_{t-1|t-1}^{(i)}) p(X_{t|t-1}^{(i)}|X_{t-1|t-1}^{(i)}, \bar{z}_t) p(X_{t-1|t-1}^{(i)})}{q(X_{t|t-1}^{(i)}|X_{t-1|t-1}^{(i)}, \bar{z}_t) q(X_{t-1|t-1}^{(i)})} \quad (24)$$

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(z_t|X_{t|t-1}^{(i)}) p(X_{t|t-1}^{(i)}|X_{t-1|t-1}^{(i)})}{q(X_{t|t-1}^{(i)}|X_{t-1|t-1}^{(i)})} \quad (25)$$

Donde se obtendrá $X_{t|t-1}^{(i)}$ de la ecuación (20) como:

$$X_{(t|t-1)}^{(i)} = f(X_{(t-1|t-1)}^{(i)}) + u_{t-1}^{(i)} \quad (26)$$

La función densidad de probabilidad $p(X_t|\bar{z}_t)$ se puede aproximar como:

$$p(X_t|\bar{z}_t) \approx \sum_{i=1}^N w_i^{(t)} \delta(X_t - X_{t|t-1}^{(i)}) \quad (27)$$

Donde la actualización de los coeficientes se realiza de forma recursiva mediante la ecuación (25). Con esto se logrará que a medida que se avance en cada iteración el algoritmo consiga converger como se puede observar en la Figura 11.

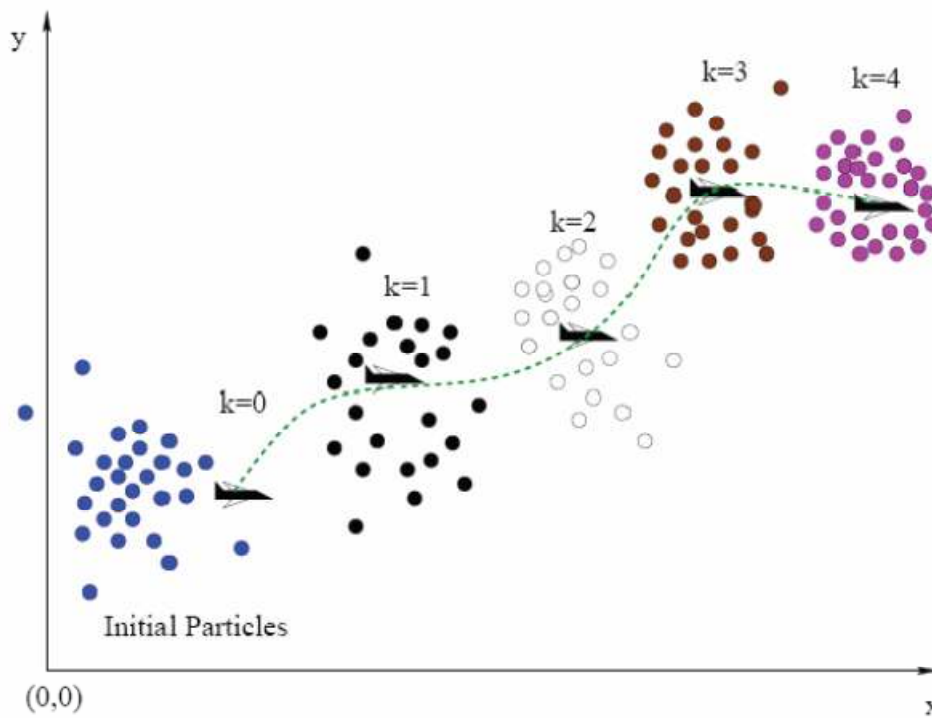


Figura 11 : Ejemplo de convergencia de un filtro de partículas.

4.1.2 Remuestreo

Este algoritmo secuencial presenta algunos problemas cuando se ejecuta varias veces la ecuación (27), debido a que las partículas se dispersan. Esto lo causa el incremento progresivo de la varianza de X_t . Para solucionarlo, [23], los coeficientes correspondientes a los $X_t^{(i)}$ que se alejen mucho del valor esperado \widehat{X}_t , deberán tender a cero, $w_t^{(i)} \rightarrow 0$. A esto se le llama problema de la degradación del filtro de partículas, y se puede ver representado en la Figura 12. El proceso que se ve en esta figura es cómo los pesos que se han calculado para cada partícula y que son mayores, es decir, más probabilidad de que el objetivo esté en esa partícula, se remuestran. En cambio las que tienen los pesos menores se eliminan, haciendo que el número de las partículas que se han eliminado sea el mismo a las partículas que se han remuestreado. Esto se hace en cada iteración, consiguiendo tener muchas réplicas de las partículas que están cerca del objetivo, y casi ninguna de las que no aportan información relevante. En ese momento el algoritmo habrá conseguido converger.

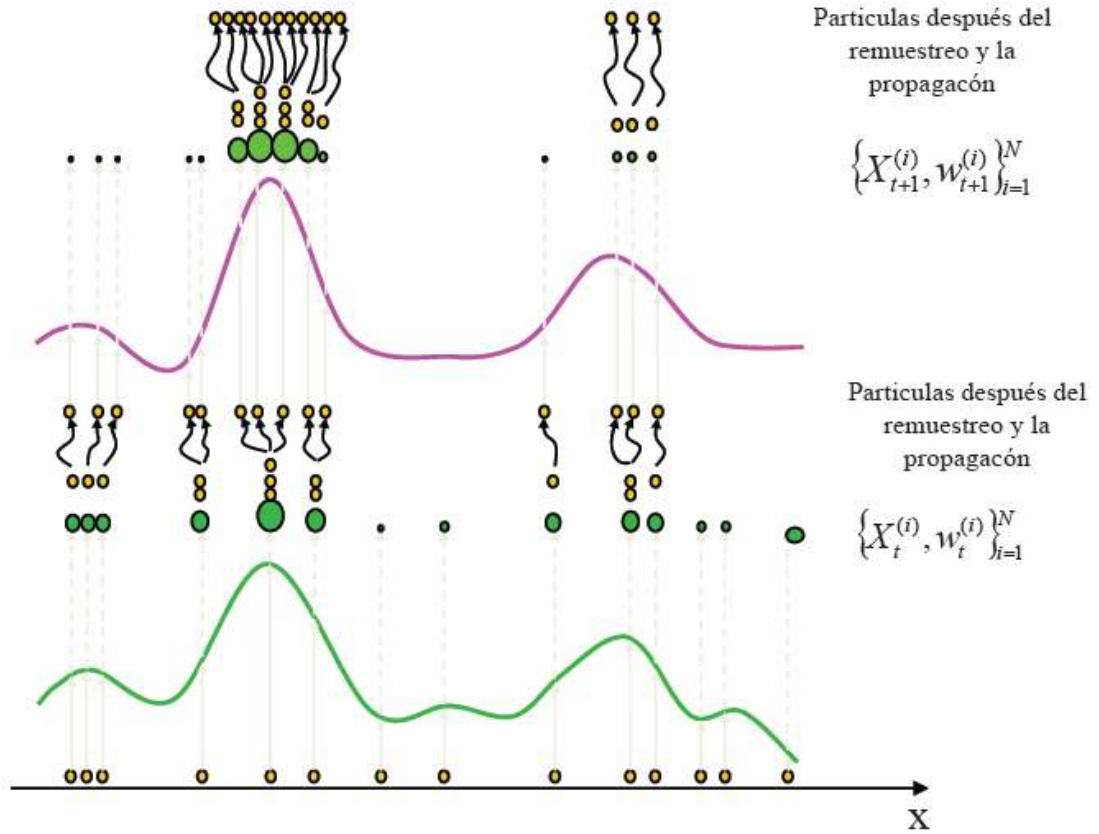


Figura 12 : Ejemplo de remuestreo de un filtro de partículas.

Para medir dicha degradación se introduce un nuevo parámetro, el número efectivo de muestras, N_{eff} , que puede calcularse como:

$$N_{eff} = \frac{1}{\sum_{i=1}^N (w_i^{(t)})^2} \quad (28)$$

Normalmente, en cada iteración se suele introducir un remuestreo, que sustituye las partículas con baja probabilidad por las que tienen alta probabilidad, manteniendo siempre el mismo número de partículas utilizadas en cada iteración. Este paso sólo se realiza cuando $\widehat{N_{eff}} \leq N$.

La utilización del remuestreo lo que permite es eliminar las partículas que tienen pesos pequeños y replicar aquéllas que tienen pesos mayores. Los pasos que se tienen que seguir para conseguir esta replicación son:

1. Obtener M partículas, $X_{t|t}^{*(m)}$ de la distribución discreta:

$$X_t = \left\{ X_{t|t-1}^{(m)}, w_t^{*(i)} \right\}_{m=1}^M$$

2. Sea $X_{t|t}^{(m)} = X_{t|t-1}^{*(m)}$, se asignan los pesos iguales a todas las partículas con valor $1/M$, Figura 13.

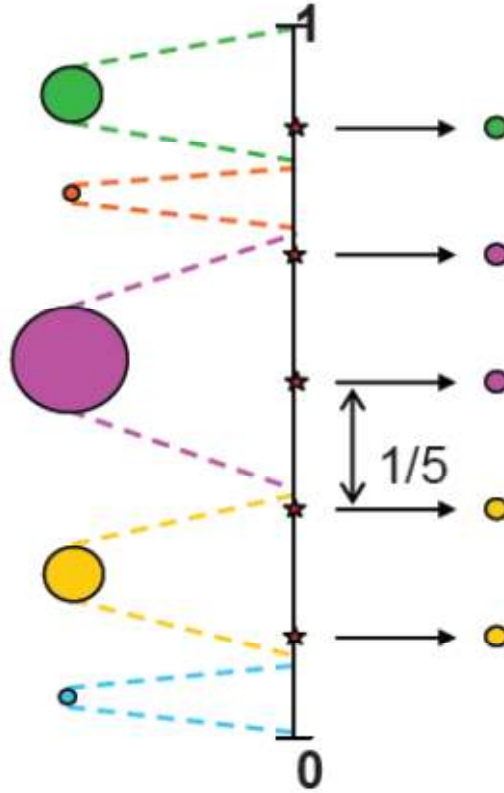


Figura 13 : Ejemplo de reconfiguración de las partículas en el remuestreo.

Esto es sólo un ejemplo de remuestreo de partículas, existen infinitud de métodos para realizar este proceso. Se ha elegido este por simplicidad, siendo el más sencillo e intuitivo.

4.1.3 Aproximación de Bootstrap

Para realizar la implementación de este algoritmo, se necesita primero poder cuantificar la probabilidad $q(X_{t|t-1}^{(i)}|X_{t-1|t-1}^{(i)})$, $p(X_{t|t-1}^{(i)}|X_{t-1|t-1}^{(i)})$ y la verosimilitud $p(z_t|X_{t|t-1}^{(i)})$. Cuando el ruido no puede ser modelado como una distribución Gaussiana, la definición de estas probabilidades es muy compleja.

La aproximación de Bootstrap lo que permite es simplificar el cálculo de las probabilidades anteriores, haciendo que:

$$q(X_{t|t-1}^{(i)}|X_{t-1|t-1}^{(i)}) = p(X_{t|t-1}^{(i)}|X_{t-1|t-1}^{(i)}) \quad (29)$$

Con este resultado se consigue eliminar dos de las probabilidades que se necesitan determinar para poder realizar el algoritmo, con lo que la ecuación de actualización de los coeficientes queda como:

$$w_t^{(i)} \propto w_{t-1}^{(i)} \frac{p(z_t | X_{t|t-1}^{(i)}) p(X_{t|t-1}^{(i)} | X_{t-1|t-1}^{(i)})}{q(X_{t|t-1}^{(i)} | X_{t-1|t-1}^{(i)})} \quad (30)$$

$$w_t^{(i)} \propto w_{t-1}^{(i)} p(z_t | X_{t|t-1}^{(i)}) \quad (31)$$

El procedimiento del filtro de partículas de Bootstrap es igual al que se había explicado anteriormente, salvo que en este caso se va a utilizar la ecuación (31) en lugar de la ecuación (25). Los pasos a seguir para la implementación de este algoritmo son:

$$\left[\{X_{t-1|t-1}^{(i)}, w_{t-1}^{(i)}\}_{i=1}^N \right] = DFP \left[\{X_{t-1|t-1}^{(i)}, w_{t-1}^{(i)}\}_{i=1}^N, z_{t-1} \right]$$

1. Se muestrea $X_{(t|t-1)}^{(i)} = f(X_{(t-1|t-1)}^{(i)}) + u_{t-1}^{(i)}$ $i = 1, 2, \dots, N$.

2. Se actualizan los coeficientes y se calcula $\widehat{N_{eff}}$:

$$w_t^{(i)} \propto w_{t-1}^{(i)} p(z_t | X_{t|t-1}^{(i)})$$

3. Si $\widehat{N_{eff}} \leq N$, entonces se realiza el remuestreo, sino se pasa al siguiente paso.

4. Se selecciona el líder y se transmite $\{X_{t|t}^{(i)}, w_t^{(i)}\}_{i=1}^N$ al nuevo líder, si la estimación se está realizando en un método que tenga líder. Si por el contrario la estimación se hace a posteriori, entonces esos datos se utilizan para la siguiente iteración.

Capítulo 5

Resultados Experimentales

En este capítulo se van a presentar los resultados que se han obtenido al implementar lo que se ha desarrollado de forma teórica. Se han elegido los resultados que son más relevantes para este estudio. Se va a explicar cómo se deben elegir los parámetros más significativos de cada técnica y se van a comparar los resultados de unas técnicas con otras. Se deducirá si hay beneficio o si por el contrario se obtienen peores resultados.

5.1 Elección de parámetros

En este apartado se va a desarrollar cuál tiene que ser el valor de cada uno de los parámetros significativos de cada técnica.

5.1.1 Elección de parámetros para CFAR

En la técnica CFAR los dos parámetros más significativos que hay que elegir son el parámetro α , que es un factor que controla la memoria que va a tener el umbral calculado de los instantes anteriores, y la constante C utilizada para el cálculo del umbral y que dependerá de la probabilidad de falsa alarma que se quiera dar al sistema.

CAPÍTULO 5 RESULTADOS EXPERIMENTALES

Se va a empezar por elegir la constante C . Esta constante está relacionada con la probabilidad de falsa alarma de cada sensor como especifica la ecuación (32). Hay que destacar que la probabilidad de falsa alarma del sistema tiene el mismo valor que la probabilidad de falsa alarma de cada sensor, en el caso de que se asigne la misma probabilidad de falsa alarma a todos los sensores, ya que los sensores actúan de forma independiente.

$$P_{fa_i} = \frac{1}{\sqrt{2\pi}} \int_C^{\infty} \exp\left(-\frac{1}{2}u^2\right) du \quad (32)$$

Para desarrollar el valor de la constante C en función de la probabilidad de falsa alarma se va a utilizar la función Q , una función muy utilizada y conocida en ingeniería, y que se define en la ecuación (33).

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^{\infty} e^{\left(-\frac{u^2}{2}\right)} du \quad (33)$$

Si se combinan ambas ecuaciones, es decir (32) y (33), se obtiene el valor de C en función de la probabilidad de falsa alarma como se especifica en la ecuación (34).

$$P_{fa_i} = Q(C) \Rightarrow C = Q^{-1}(P_{fa_i}) \quad (34)$$

La constante C por tanto dependerá del valor que se le dé a la probabilidad de falsa alarma. En el caso de este tipo de redes la probabilidad de falsa alarma que se da a los sensores está en torno a un valor de 0.01, que será el valor que se va a utilizar en este caso.

El segundo parámetro que se va estudiar es el parámetro α , o también conocido como el factor de memoria. Este factor lo que hace es indicar cuánto de los instantes anteriores recuerda este umbral para los instantes posteriores. En las siguientes figuras se va a mostrar el resultado de aplicar este método para calcular el umbral. La potencia recibida de los sensores en este caso es una señal cuadrada de amplitud 4, y activo el 50% del ciclo de la señal. En la Figura 14, la Figura 15 y la Figura 16 se puede ver cómo el umbral se adapta a una señal de potencia recibida en función del valor de la variable α .

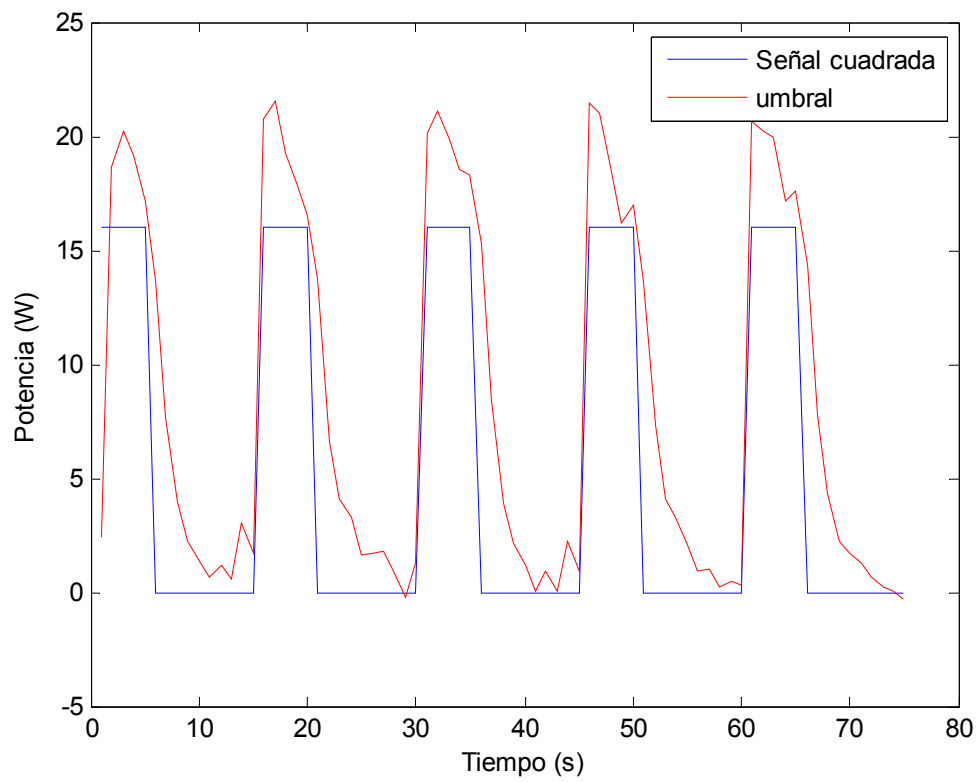


Figura 14 : Resultado para un valor de $\alpha = 0.3$.

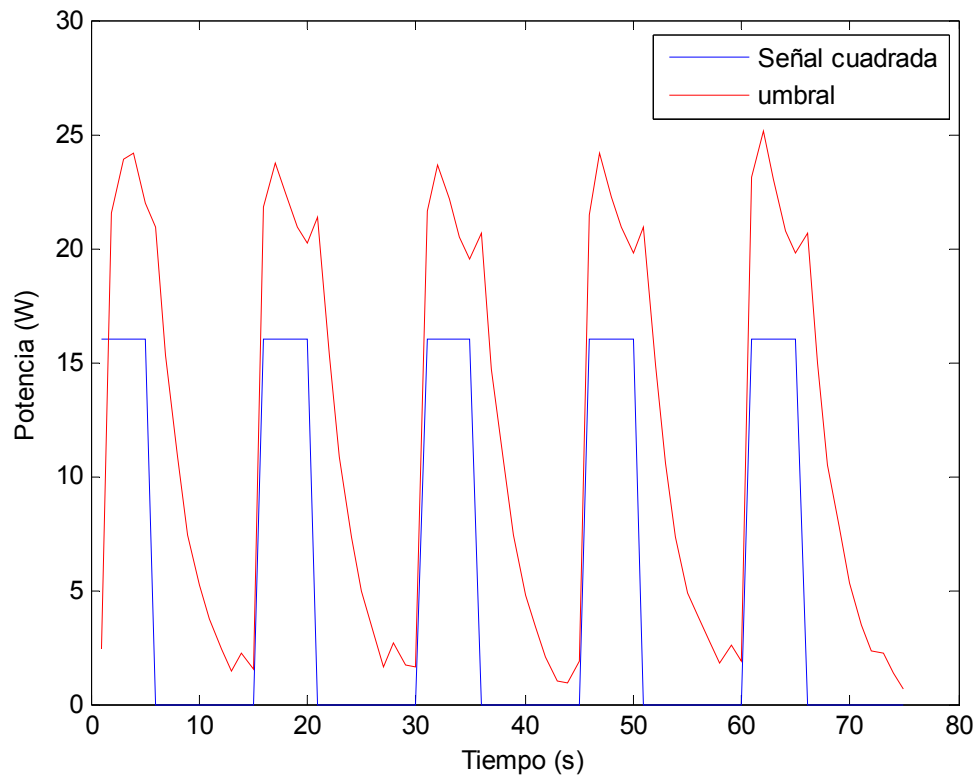


Figura 15 : Resultado para un valor de $\alpha = 0.5$

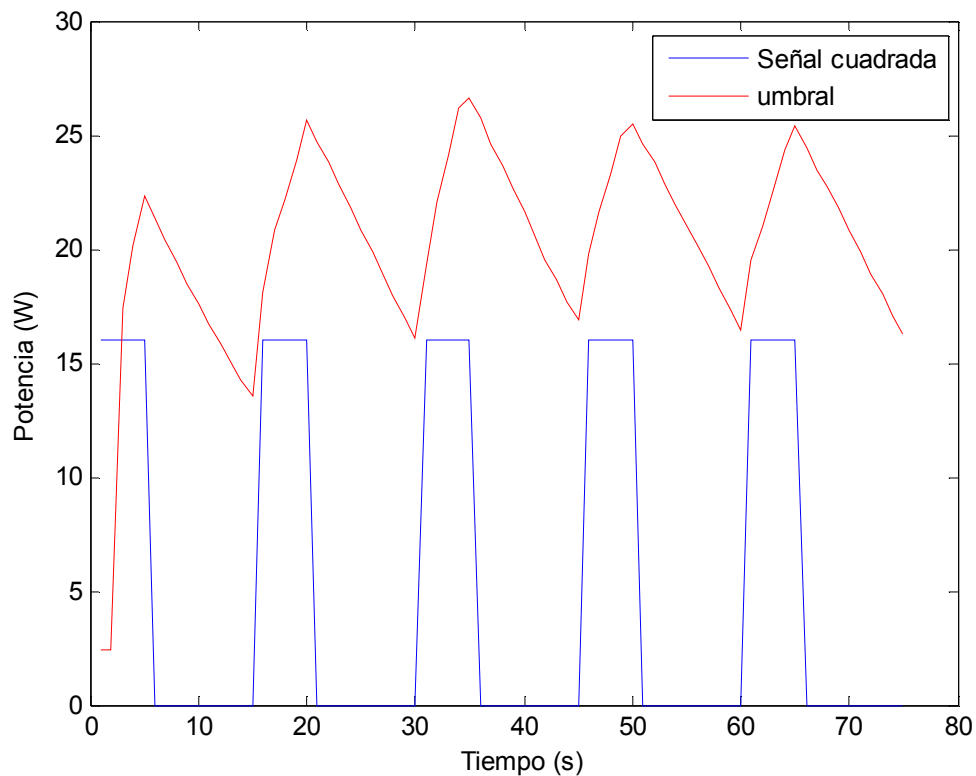


Figura 16 : Resultado para un valor de $\alpha = 0.9$

Después de observar estos resultados se puede ver que cuando el valor de α es muy pequeño, el umbral prácticamente sigue a la señal de la potencia recibida, en cambio cuando el valor de α es muy grande lo que pasa es que el umbral no es capaz de alcanzar otra vez a la señal de la potencia recibida, por lo que cada vez se aleja más. Esto se debe a que el factor α lo que hace es indicar cuánto de los instantes anteriores se recuerda para calcular el umbral en el instante posterior. Si se quiere que el sistema tenga en cuenta muchos de los datos anteriores entonces el valor de α será alto, si lo que se quiere es lo contrario, entonces el valor de α debe ser pequeño. En este caso, lo que se quiere es tener un aprendizaje de los datos anteriores, pero a su vez, en el momento en el que se cambia de detectar a no detectar, o viceversa, cuando el valor de la potencia recibida tiene un cambio grande, lo que se quiere es que el umbral también se adapte a este cambio, por lo que habrá que utilizar un valor de α intermedio, ya que se quieren conseguir los dos fenómenos. En este caso se ha elegido un valor de 0.65.

5.1.2 Elección de Parámetros para Fusion Decision Rule.

Al igual que ocurría con la técnica CFAR será necesario elegir el valor de C . El desarrollo es análogo al realizado en el apartado anterior, y el valor de C dependerá directamente de la probabilidad de falsa alarma individual que se fije para los sensores, tal y como se indica en la ecuación (34).

Puesto que lo que se pretende es comparar dichos métodos en igualdad de condiciones para ambos casos se elegirá el mismo valor para la probabilidad de falsa alarma y dicho valor se fijará a 0.01. La constante C se calculará a partir de este valor.

Finalmente para el método de Decision Fusion Rule con aproximación será necesario fijar el valor del umbral T . En apartados anteriores se ha visto que en este caso Λ se calculaba como la suma de las decisiones individuales, $I_i = \{0,1\}$ ($i = 1, \dots, N$). De forma que se podrá ver el umbral T , como el número de decisiones positivas, $I_i = 1$, que es necesario tener para considerar que efectivamente se ha detectado algo. Para el escenario que se está analizando se considerará que sería suficiente con que la mitad de las decisiones tomadas fueran positivas, para decidir que existe un objeto dentro del área de vigilancia. Hay que recordar que el nodo líder encargado de realizar la decisión, tendrá acceso a su información y a la de todos sus vecinos, con lo que si más de la mitad de los sensores de ese grupo deciden de forma positiva, la decisión global que tomará el nodo líder será positiva.

5.2 Prestaciones

En este apartado se van a presentar los distintos resultados que se han obtenido al implementar el sistema. Hay que indicar que los resultados que se han promediado sobre 25 simulaciones para obtener conclusiones más generales acerca de su funcionamiento. Tanto la implementación como las simulaciones se han realizado en Matlab.

Los resultados que se presentan realizarán una comparación de los métodos explicados en el Apartado 3.3 y un método utilizado como referencia. Dicha referencia consiste en la técnica de detección más simple, aquella que impone un umbral fijo a todos los sensores independientemente de lo que se esté midiendo o de la información que puedan aportar otros sensores de la red. De este modo, el umbral de detección para la técnica de umbral fijo vendrá dado por la ecuación (35).

$$\Lambda_{fijo} = C = Q^{-1}(P_{fa_i}) \quad (35)$$

Cuyo valor coincide con el valor de C analizado en el apartado anterior para las otras técnicas de detección. El método de referencia se ha implementado para los dos tipos de procesamiento: centralizado y distribuido.

Resulta interesante recordar que algunas técnicas de detección no son aplicables para todos los modelos de procesamiento, tal y como se indicó en el Apartado 3.3. Debido a esto, los resultados mostrados para cada modelo de procesamiento son:

- Procesamiento centralizado:
 - Técnica CFAR
 - Técnica de Umbral Fijo
- Procesamiento distribuido:
 - Técnica Decision Fusion Rule (DFR) \Rightarrow En sus dos implementaciones: óptima y aproximación.
 - Técnica de Umbral Fijo

5.2.1 Números medio de vecinos

Además de estos modelos de procesamiento comentados en el apartado anterior, se ha considerado interesante analizar distintos tipos de distribuciones de sensores: una aleatoria y una determinista.

Para ambas distribuciones se desplegará el mismo número de sensores, en un intento de realizar una comparación más equitativa de ambas. En el Apartado 2.2 se pudo ver un ejemplo real de las redes utilizadas para esta implementación, con un total de 100 sensores desplegados en un área de 100 x 100 metros. Para el caso del despliegue aleatorio se seguirá una distribución uniforme en todo el área y se tendrá una densidad de sensores de $0.01 \text{ sensores}/m^2$. En el caso determinista, los sensores se colocarán siguiendo una rejilla y con una separación fija entre ellos de 10 metros. El hecho de desplegar una red con un número de sensores que puede ser considerado elevado, se debe a que el modelo de medida utilizado tiene una fuerte caída con la distancia, con lo que ha sido necesario disminuir la distancia entre sensores, aumentando así el número de sensores que componen la red.

Una diferencia importante entre ambos métodos y que puede afectar a los resultados que se obtengan, es el hecho de que en el despliegue determinista cada sensor tiene un número fijo de vecinos, mientras que en el aleatorio este número puede variar. En la Tabla 2, se puede ver el número medio de vecinos obtenido en las simulaciones realizadas. Como se puede apreciar estos números son muy similares puesto que el número de sensores que componen la red es elevado, y además los resultados están promediados para varias simulaciones.

Tabla 2 : Número medio de vecinos promediado con varias iteraciones.

Despliegue rejilla	Despliegue aleatorio
7	8

5.2.2 Probabilidad de falsa alarma y de detección

En este apartado se van a comparar las probabilidades de falsa alarma y las probabilidades de detección para cada una de las técnicas con los distintos procesamientos de red y los distintos despliegues estudiados.

Por un lado, la probabilidad de falsa alarma nos va a dar una idea de cuánta probabilidad hay de que el sistema afirme que ha detectado algo, cuando en realidad lo único que existía era ruido. Por el otro, la probabilidad de detección nos va a determinar con qué probabilidad se producen detecciones del objetivo cuando éste tiene que ser detectado.

❖ Procesamiento centralizado

En la Tabla 3 se muestran los resultados de las probabilidades de falsa alarma y de detección para el caso de tener una red con procesamiento centralizado. En la misma tabla se muestran los resultados para un despliegue en forma de rejilla y un despliegue aleatorio. Las técnicas que se están comparando son la técnica CFAR y la técnica de umbral fijo que son las que se han implementado para este tipo de procesamiento de la red.

Tabla 3 : Probabilidad de falsa alarma y de detección para una red centralizada.

	Despliegue rejilla		Despliegue aleatorio	
	Pfa	Pd	Pfa	Pd
Técnica CFAR	0.0163	0.8439	0.0164	0.8371
Técnica umbral fijo	0.0325	0.8329	0.0322	0.8373

En esta tabla se puede observar cómo la probabilidad de falsa alarma es mucho mayor para el caso de la técnica del umbral fijo, por lo que se puede decir, que la técnica CFAR da muchas menos falsas alarmas y por tanto se equivocarán en menos casos que la técnica del umbral fijo. Este resultado se puede observar en ambos tipos de despliegue. Por lo que en este caso, el cambio de técnica se podría considerar ya que los beneficios obtenidos serían bastantes, y por este motivo el sistema funcionaría considerablemente mejor.

❖ Procesamiento distribuido o colaborativo

En la Tabla 4 se muestran los resultados de las probabilidades de falsa alarma y de detección para el caso de tener una red con procesamiento distribuido. En la misma tabla se muestran los resultados para un despliegue en forma de rejilla y un despliegue aleatorio. Las técnicas que se están comparando son la técnica de umbral fijo y las dos variantes de la técnica Decison Fusion Rule, que son las que se han implementado para este tipo de procesamiento de la red.

Tabla 4 : Probabilidad de falsa alarma y de detección para una red distribuida.

	Despliegue rejilla		Despliegue aleatorio	
	Pfa	Pd	Pfa	Pd
Técnica umbral fijo	0.0338	0.8317	0.0337	0.8365
Técnica DFR	0.0171	0.9681	0.0170	0.9663
Técnica DFR Aproximación	0.0336	0.8343	0.0336	0.8387

En esta tabla se puede observar cómo en el caso de la técnica Decision Fusion Rule en su variante óptima, los resultados que se obtienen son mejores que en los otros dos casos. En estos resultados se puede observar que en cuanto a la probabilidad de falsa alarma y de detección, la técnica aproximada de Decision Fusion Rule no obtiene beneficios, por lo en este ámbito, no merece la pena cambiar la técnica sencilla del umbral fijo por esta aproximación. En el caso óptimo de dicha técnica los beneficios son bastante mayores, por lo que si interesaría cambiar la técnica ya que los resultados obtenidos harían que el sistema mejorara considerablemente.

5.2.3 Número de sensores activos

Tal y como se había comentado la detección también permite ahorrar energía puesto que se limitará el número de sensores que necesitan realizar comunicaciones o procesar datos. Debido a esto se ha decidido realizar un estudio del número de sensores que aportan información para la posterior estimación de un blanco, es decir, los sensores que permanecen activos.

❖ Procesamiento centralizado

En primer lugar se va a realizar el estudio de este resultado para una red centralizada, es decir, una red donde hay una estación base a la que se le comunican los resultados y es la que hace el procesamiento de los mismos para calcular la estimación. Donde todos los sensores se despiertan en cada instante de tiempo para decidir si han medido o no. En estas gráficas se hace el recuento de los sensores que detectan algo en cada instante de tiempo.

En la Figura 17 se comparan el número de sensores activos medio para una red centralizada, con un despliegue en forma de rejilla, obtenidos para las técnicas CFAR y la técnica del umbral fijo, que son las técnicas que se han implementado para este tipo de procesamiento.

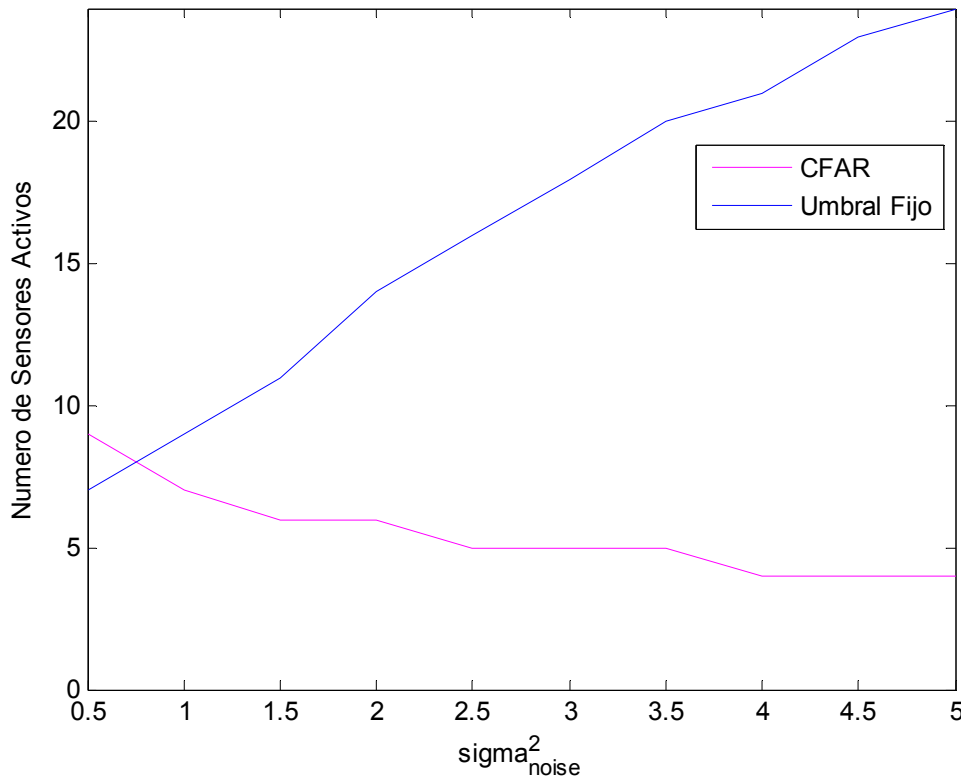


Figura 17 : Número de sensores activos medio en una red centralizada con despliegue en rejilla.

En esta figura se puede observar que para el caso de la técnica del umbral fijo se obtienen peores resultados, la diferencia entre ambas técnicas es notable cuando el ruido va aumentando. Esto se debe a que la técnica CFAR calcula el umbral en función de cómo va cambiando el entorno, al contrario que la técnica del umbral fijo, que desde que se implanta el sistema se asigna un umbral que no va a cambiar. Por lo que si el sistema se va a instalar en un entorno en el que no se conoce de antemano el valor exacto que tendrá el ruido, cosa que suele suceder muy a menudo, es mejor que se instale la técnica CFAR puesto que se va a adaptar a cualquier cambio que se produzca en el ambiente, en cambio la técnica del umbral fijo no lo hará.

En la Figura 18 se va a observar el mismo resultado que en el caso anterior pero para un despliegue aleatorio. Se puede observar que el resultado en términos generales es el mismo, ya que sólo se ha cambiado la forma de despliegue de la red, para ver que aunque sea aleatorio, con la técnica CFAR se sigue ganando. Esto se debe a que con el despliegue en forma de rejilla no se pierde generalidad, aunque sea aleatorio, el resultado sigue siendo el mismo.

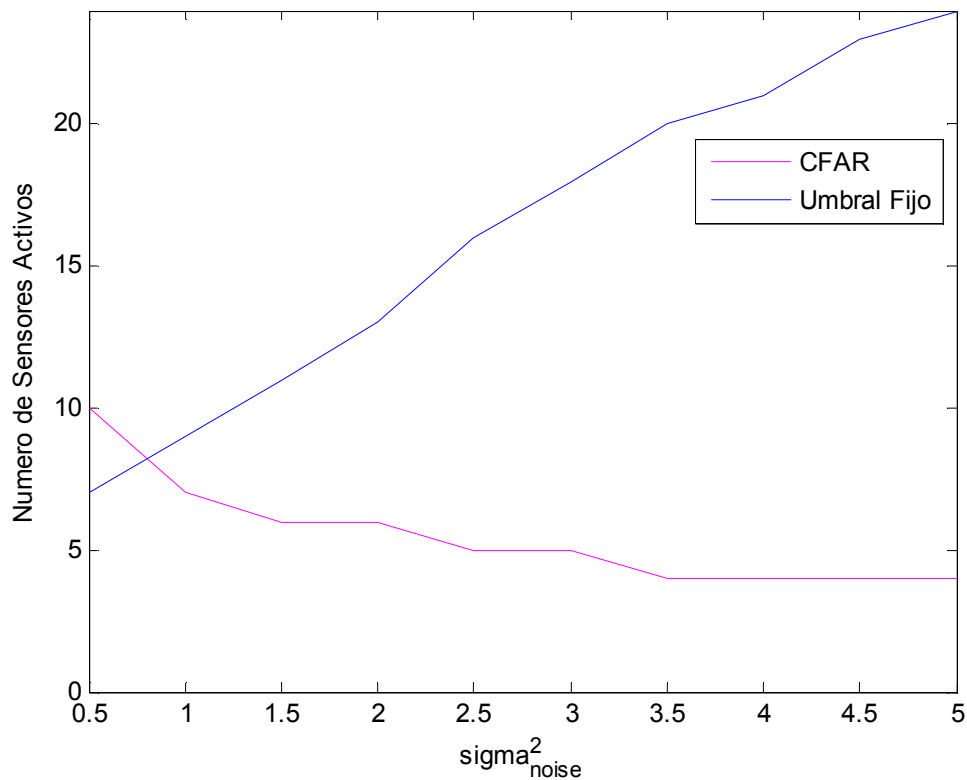


Figura 18 : Número de sensores activos medio para una red centralizada con despliegue aleatorio.

❖ Procesamiento distribuido o colaborativo

Ahora se va a proceder a hacer las comparaciones para una red distribuida, es decir, en la que se tiene un líder que hace el procesamiento de los datos y va cambiando en función de cómo se mueve el objetivo. Por lo que sólo se despertarán los nodos que son líder y los nodos vecinos. En estas gráficas se hace el recuento de los sensores que se despiertan y determinan que han detectado algo. En este caso se obtienen números de sensores activos que están limitados por el número de vecinos debido al tipo de procesamiento.

Se va a comparar ahora la técnica Optimal Decision Fusion Rule y su aproximación, con la técnica del umbral fijo. En la Figura 19 se van a comparar los números de sensores activos medio para estas tres técnicas en el caso de tener una red distribuida con despliegue en forma de rejilla.

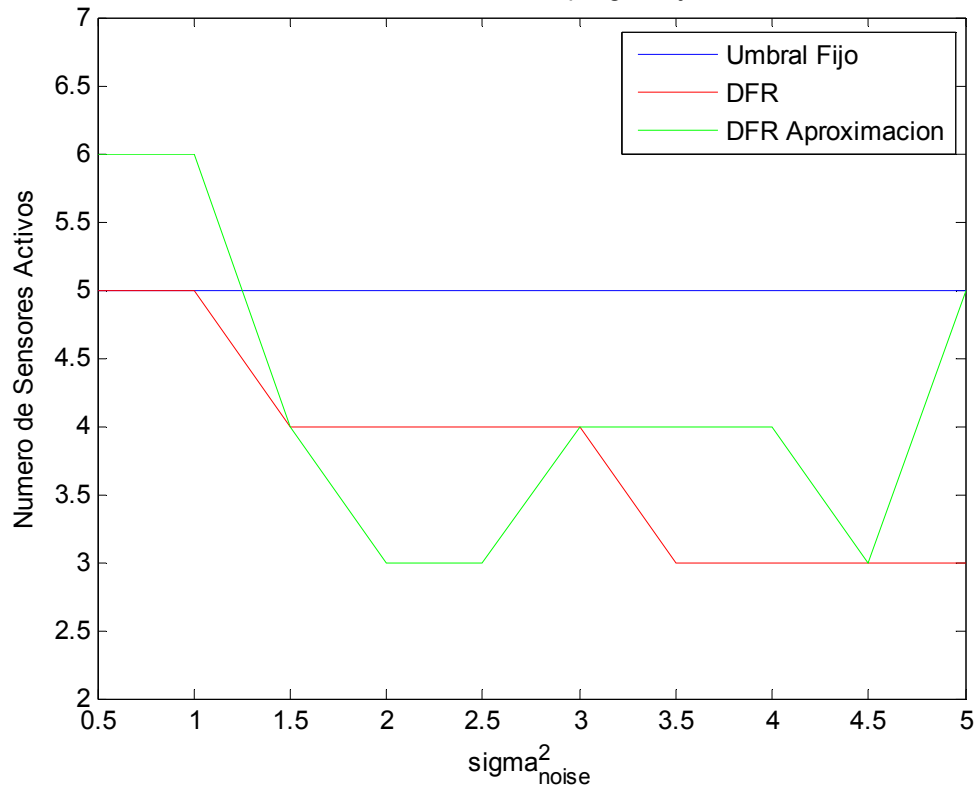


Figura 19 : Número de sensores activos medio para una red distribuida con despliegue en rejilla.

En esta figura se puede detectar que cuando el ruido aumenta, el caso de la técnica de umbral fijo tiene mayor número de sensores activos que las dos variantes de la técnica Decision Fusion Rule. Dentro de las dos variantes se puede observar que obtienen resultados muy similares, aunque cuando el ruido aumenta, el caso óptimo funciona mejor a cambio de aumentar el procesamiento en el sensor. En definitiva, conviene implementar la técnica Decision Fusion Rule en lugar de la técnica del umbral fijo, ya que se obtienen mejores beneficios. Dentro de las dos variantes, sería mejor implementar la óptima, puesto que cuando el ruido aumenta funciona mejor, y si no se sabe cómo va ser el ambiente del sistema, es mejor gastar un poco, ya que la diferencia de cómputo no es tan grande, en relación a los beneficios obtenidos y además se consigue un sistema más estable.

En la Figura 20 se va a observar el mismo resultado que en el caso anterior pero sólo se va a variar la forma de despliegue de la red, en lugar de ser en forma de rejilla, será de forma aleatoria. En esta figura se puede ver que los resultados obtenidos siguen cumpliendo lo mismo que se ha dicho para el caso anterior.

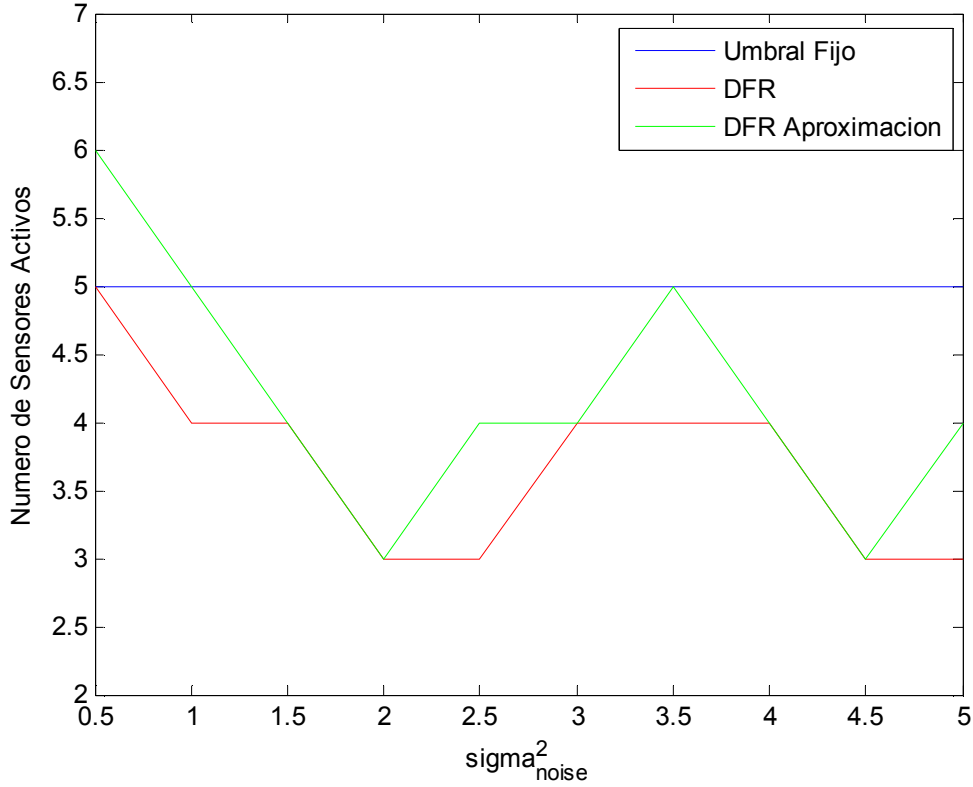


Figura 20 : Número de sensores activos medio para una red distribuida con despliegue aleatorio.

En el caso de una red centralizada, las mejoras obtenidas con la técnica CFAR son evidentes. Para el caso de una red distribuida, en realidad las diferencias no son tan obvias, puesto que los vecinos próximos a cada sensor son casi siempre los mismos, aunque ya se ha visto que las dos variantes de la técnica Decision Fusion Rule consiguen mejorar los resultados obtenidos en el caso del umbral fijo, por lo que en este ámbito la técnica del umbral fijo queda también descartada.

5.2.4 Prestaciones en Localización

Hasta ahora se ha visto los resultados sólo desde el punto de vista de la detección, pero se debe tener en cuenta que se está aplicando a un sistema de localización, cuyo último fin es obtener una buena estimación de la posición del blanco. Para ello se implementó un filtro de partículas, siguiendo los pasos indicados en el Apartado 4.1.

Para comprobar las prestaciones en localización se estudiará el error cuadrático medio (RMSE) obtenido con cada una de las técnicas en distintos escenarios (ya sea con procesamiento distribuido o centralizado). Dicho error se calcula según la ecuación (36).

$$RMSE(\hat{x}) = \sqrt{MSE(\hat{x})} = \sqrt{E((\hat{x} - x)^2)} \quad (36)$$

Siendo \hat{x} la estimación y x la posición real del blanco.

❖ Procesamiento centralizado

En primer lugar se van a comparar los resultados del estimador obtenidos para una red centralizada en el caso de despliegue en forma de rejilla. En la Figura 21 se pueden observar los resultados para la técnica CFAR y la técnica del umbral fijo, que son las técnicas que se han implementado para este tipo de procesamiento.

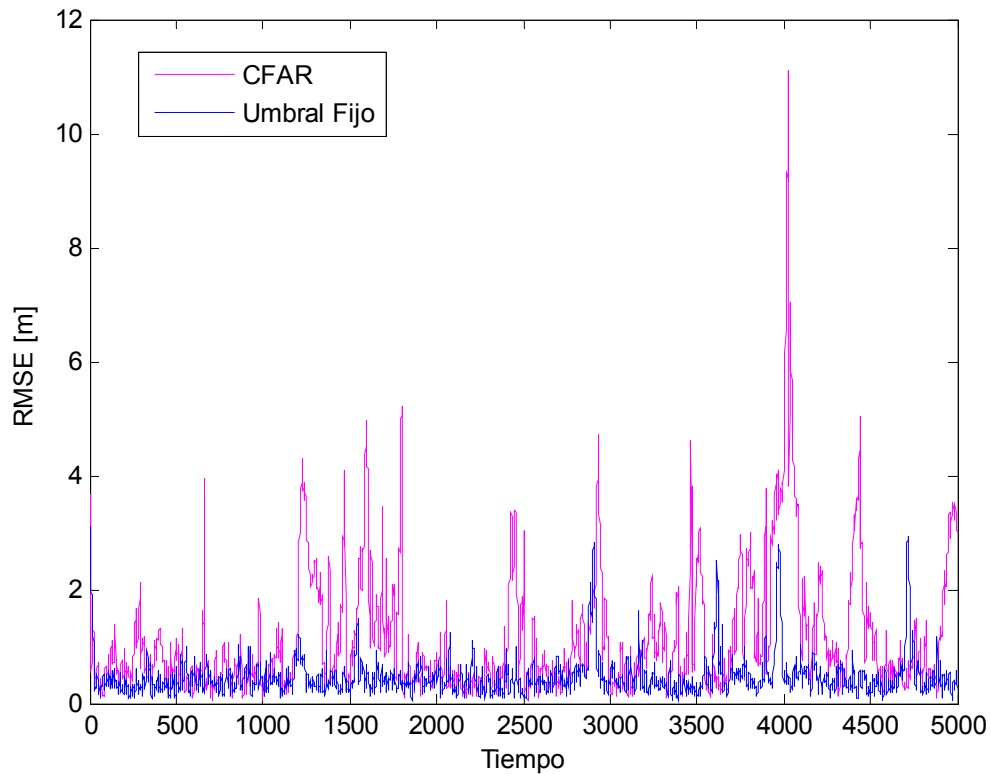


Figura 21 : RMSE obtenido para una red centralizada con despliegue de rejilla.

En esta figura se puede observar cómo ambos métodos obtienen resultados similares en cuanto al RMSE. La principal diferencia reside en que las estimaciones para cada método están hechas en base a un número muy distinto de medidas. Para el caso de la técnica CFAR este número es mucho menor cuando el ruido aumenta, tal y como se ha visto en el apartado anterior. La pequeña diferencia que hay entre esta técnica y la técnica del umbral fijo, se debe a que la estimación se hace con un menor número de sensores y por ello puede que al ser más selectivo obtenga un error ligeramente mayor. Esto no es ningún inconveniente porque ese pequeño aumento en el error, no es comparable a lo que se obtiene al activar un número bastante menor de sensores.

En la Figura 22 se van a presentar los mismos resultados que en el caso anterior pero utilizando un despliegue de la red aleatorio. Se puede comprobar que todavía se sigue observando el mismo resultado que en el caso anterior, aunque en este caso los valores de ruido son más altos. Este aumento en el error se debe a que como la red es aleatoria, existirán zonas donde haya menor número de sensores y por tanto hagan que el error aumente, ya que en esas zonas se perderá bastante precisión.

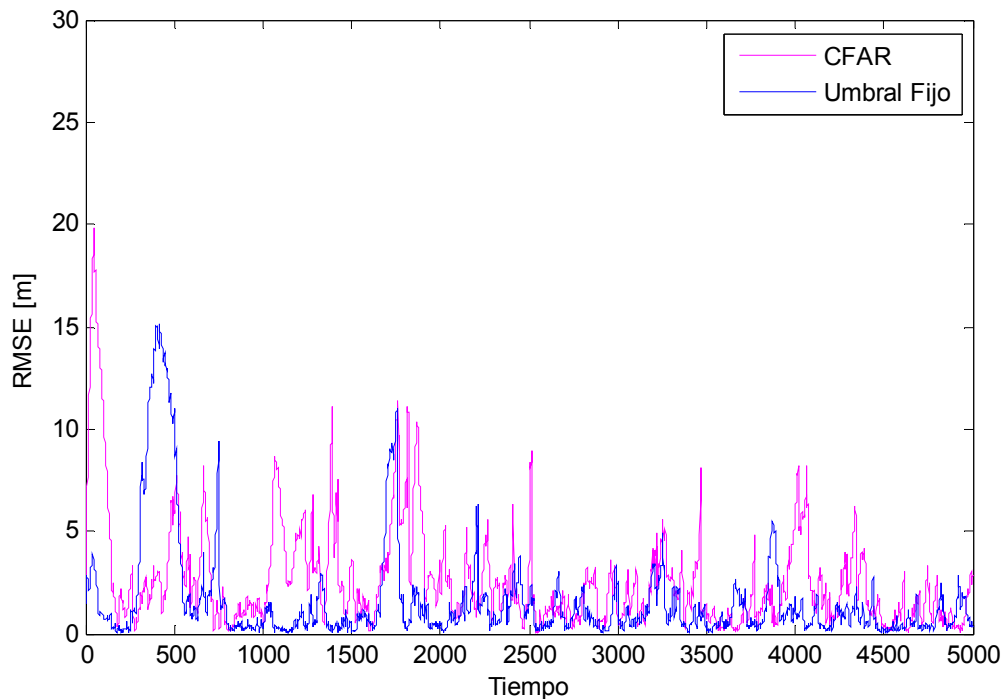


Figura 22 : RMSE para una red centralizada con despliegue aleatorio.

Se puede afirmar que aunque en cuanto al error RMSE no se observan muchos cambios, en el procesamiento centralizado, la técnica CFAR obtiene mejores resultados que la técnica del umbral fijo, debido a los demás criterios. Por esta razón se podría decir que la técnica CFAR tiene un funcionamiento mejor a la técnica del umbral fijo en un procesamiento centralizado.

❖ Procesamiento distribuido o colaborativo

Ahora se va a proceder a comparar los resultados obtenidos para el estimador utilizado en el caso de tener una red distribuida, tanto para el caso de tener un despliegue en forma de rejilla como uno aleatorio.

En la Figura 23 se van a comparar los resultados obtenidos del estimador para las dos variantes de la técnica de Decision Fusion Rule, con la técnica del umbral fijo. El caso en el que se han aplicado estas técnicas es para una red distribuida con despliegue en forma de rejilla.

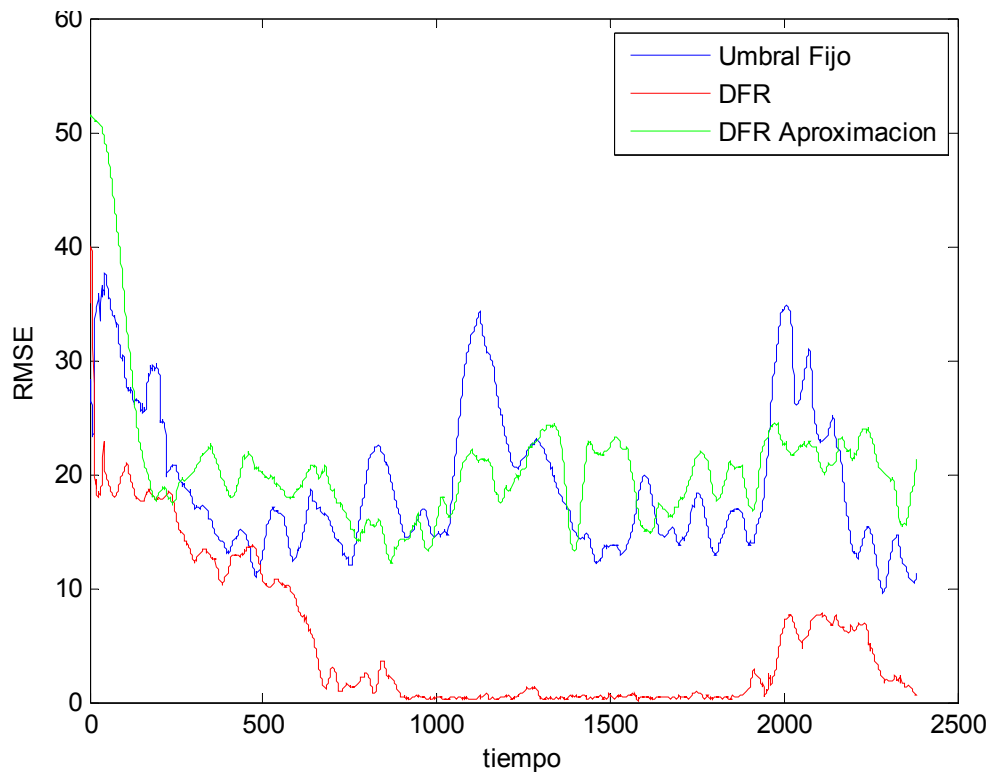


Figura 23 : RMSE para una red distribuida con despliegue en forma de rejilla.

En esta figura se puede observar que si se comparan los resultados obtenidos para las dos variantes de la técnica Decision Fusion Rule, con la técnica del umbral fijo, se obtienen errores bastante menores. Cabe destacar, que en el caso de la aproximación de la técnica Decision Fusion Rule, la diferencia con la técnica de umbral fijo cuando el ruido aumenta no es tan grande, obteniendo resultados similares. Esto se debe a que al querer que el cómputo en los sensores sea más sencillo se pierde precisión, llegando incluso a considerar, que los beneficios obtenidos no son mayores a los costes que se han reducido. Por este motivo, para la técnica con la que de verdad se obtienen grandes beneficios es para la variante óptima de la técnica Decision Fusion Rule.

En la Figura 24 se van a comparar los resultados de las mismas técnicas que en el caso anterior, pero para un despliegue aleatorio.

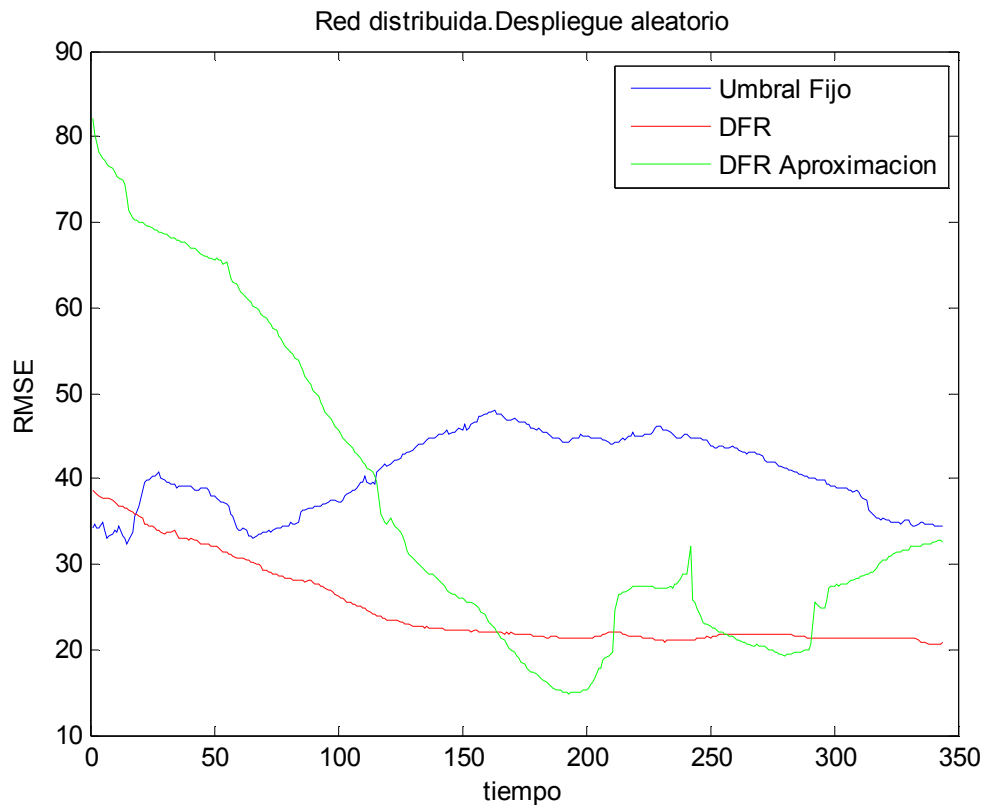


Figura 24 : RMSE para una red distribuida con despliegue aleatorio.

En esta figura lo primero que salta a la vista, es que tiene una duración del eje de tiempo menor. Esto se debe a que en realidad hay muy pocos instantes de la simulación en los que se ha decidido que se está detectando. Se cree que esto es un problema debido a que el valor de la constante T (especificada en el primer apartado de este capítulo) se fijó con el despliegue en forma de rejilla. Debido a esto, puede ser que ese valor del umbral no valga para este tipo de despliegue, pero al querer compara ambos casos con el mismo valor, se ha obtenido esto.

Si se observan sólo estos instantes de tiempo, se puede ver que el resultado que se ha destacado en el caso anterior se sigue manteniendo. Pero esto no se puede considerar muy fiable puesto que no se ve la evolución del RMSE a lo largo del tiempo, sino sólo en los primeros instantes.

Con todo esto, se puede decir que la técnica Decision Fusion Rule obtiene mejores resultados que el caso de la técnica del umbral fijo para ambos despliegues. Aunque dentro de las dos variantes de esta técnica, se ha observado que la aproximación no obtiene grandes beneficios, por lo que en este ámbito sería mejor considerar la opción óptima de dicha técnica, aunque se produjera un mayor cómputo en los sensores, ya que los beneficios obtenidos son mucho mayores y merece la pena el esfuerzo.

Capítulo 6

Conclusiones

Finalmente, en este capítulo se indicará cuáles son las conclusiones tras el estudio de las distintas técnicas de detección para la localización de objetivos realizado en este proyecto.

6.1 Conclusiones

En primer lugar como se ha mencionado en varias ocasiones, este análisis ha dejado clara la importancia de la detección dentro de cualquier aplicación, y en particular para las aplicaciones de localización. Se ha visto cómo una mala elección de la técnica o del umbral de decisión puede llevar a unas malas prestaciones del sistema en sí. Debido a ello se considera muy recomendable analizar el tipo de aplicación que se va a implementar y elegir la técnica de detección más adecuada para ella.

A lo largo de este proyecto se ha analizado dos tipos de procesamiento de datos o de señales existentes en redes de sensores. Cada uno tiene una serie de ventajas e inconvenientes que se necesitarán valorar a la hora de elegir el procesamiento que más convenga según el presupuesto, el tamaño de la red, etc. No se puede afirmar que uno sea mejor que otro puesto que dependerá de las condiciones y del escenario en el que se vayan a aplicar. Por esta razón se optó por presentar una serie de técnicas de detección para cada uno de estos tipos de procesamiento y permitir así que el usuario decidiera cual era más adecuado para su problema.

Para dar una mayor generalidad al análisis se incluyeron en el estudio distintos tipos de despliegues de la red. Los resultados obtenidos en cada caso llevan a la conclusión de que para redes grandes ambos métodos nos proporcionan prestaciones, más o menos, similares. Con lo que la elección de un modelo de despliegue u otro vendrá dada por el escenario en que se encuentre la aplicación (accesibilidad, presupuesto, etc).

El modelo de sensor utilizado en este proyecto era el propuesto en la literatura para las técnicas de detección analizadas. A la vista de los resultados obtenidos, se puede decir que quizás sería conveniente realizar pruebas con otros sensores, puesto que estos presentan una atenuación muy fuerte con la distancia, con lo que ha sido necesario disminuir la distancia entre sensores para conseguir que éstos realizaran unas medidas con la suficiente información como para estimar la posición del blanco. De todos modos, la elección de un elemento tan importante como el tipo de sensor a emplear, requiere de un análisis pormenorizado de las características del escenario o de la aplicación para la cual se vayan a utilizar.

En cuanto a las técnicas de detección analizadas, se ha podido comprobar como el modelo de referencia que fijaba un umbral fijo para cada sensor proporcionaba peores prestaciones: número de sensores activos más elevado, error en la estimación elevado, etc. La técnica CFAR aplicada al procesamiento centralizado ha permitido reducir notablemente el número de sensores activos, y en consecuencia la cantidad de mensajes enviados hacia la estación base. En el caso del procesamiento distribuido, las distintas variantes de la técnica Decision Fusion Rule permiten reducir el número de sensores activos mientras que los errores de localización no se ven incrementados. Cabe mencionar que la mejora obtenida en el caso de la aproximación para este método, no es tan notable como en el caso óptimo, con lo que deberá valorarse si es conveniente implementar la técnica en su variante óptima o aproximada.

La utilización de un filtro de partículas como algoritmo de localización ha resultado acertada puesto que se pretendía implementar un algoritmo no muy costoso computacionalmente y que proporcionara unas buenas prestaciones. Aún así, podrían probarse otros algoritmos de localización que permitirán comprobar el correcto funcionamiento del sistema de localización completo.

Finalmente, en este proyecto se han estudiado varias técnicas de detección y se han explicado las bases de cada una de ellas. Los resultados nos han permitido decantarnos por unas técnicas u otras dentro del escenario que se estaba analizando. A pesar de esto, la última decisión sobre la técnica más adecuada deberá hacerse en función del escenario, presupuesto, prestaciones exigidas, etc.

6.2 Futuras líneas de Trabajo

Como líneas futuras para este trabajo cabe destacar:

- Modelo de Sensor \Rightarrow Sería interesante variar el modelo de sensor empleado para comprobar como evolucionan las prestaciones del sistema. Este cambio podría permitir una reducción del número de sensores que componen la red, siempre que su modelo de medida lo permita.
- Estudio del Consumo Energético \Rightarrow En este proyecto se ha analizado el ahorro energético, pero sólo desde el punto de vista de los sensores que hay activos en cada instante. Para corroborar estos resultados sería conveniente realizar un estudio del consumo energético que se tiene con cada una de las técnicas de detección analizadas.
- Algoritmos de Localización \Rightarrow Como ya se había indicado, podrían realizarse pruebas con otros algoritmos de localización para ver cómo evolucionan las prestaciones del sistema con cada uno de ellos.

Capítulo 7

Presupuesto

En este capítulo se van a desarrollar todos los costes que han supuesto realizar este proyecto, detallando cada uno de ellos para llegar al presupuesto total del proyecto.

7.1 Planificación

En primer lugar, se va a desarrollar la planificación que se ha seguido para la elaboración de este proyecto. En concreto se van a explicar las distintas fases de las que consta el desarrollo de dicho proyecto, indicando la duración de cada una de ellas y por tanto calculando la duración total de todo el proyecto.

Las fases de las que consta el proyecto son:

- Fase 0: Estudio de lo que es una red de sensores y de los distintos tipos de redes que existen, así como el estudio también de algún tipo de sensor.
- Fase 1: Estudio de las distintas técnicas de detección así como lo que es la detección en sí.
- Fase 2: Estudio del algoritmo de localización que se va a utilizar para este sistema.
- Fase 3: Implementación de la red de sensores en forma de rejilla.

- Fase 4: Implementación de las técnicas de detección cada una por separado y para los dos tipos de procesamiento de la red.
- Fase 5: Prueba del sistema y obtención de los resultados.
- Fase 6: Implementación de la red de sensores con despliegue aleatoria e integración de las técnicas a dicha red.
- Fase 7: Prueba del sistema y obtención de los resultados.
- Fase 8: Realización de la memoria y de la presentación de dicho proyecto.

En la Tabla 5 se va a mostrar la duración de cada fase del proyecto así como su fecha de inicio y de fin. Tanto el día de fecha de inicio como el día de fecha de fin están incluidos en el proceso de elaboración de este proyecto. En este cálculo no se han contado los días festivos ni los días de vacaciones ni los fines de semana.

Tabla 5 : Planificación del proyecto. Especificación de la duración de sus fases.

	Fecha inicio	Fecha fin	Duración
Fase 0	21/06/2011	24/06/2011	4 días
Fase 1	27/06/2011	08/07/2011	10 días
Fase 2	11/07/2011	14/07/2011	4 días
Fase 3	15/07/2011	21/07/2011	5 días
Fase 4	22/07/2011	28/09/2011	25 días
Fase 5	29/09/2011	26/10/2011	20 días
Fase 6	27/10/2011	03/11/2011	4 días
Fase 7	04/11/2011	24/11/2011	15 días
Fase 8	25/11/2011	12/12/2011	10 días

La duración total del proyecto es de 97 días. Si la media de dedicación a este proyecto ha sido de 6 horas diarias, entonces el número total de horas para este proyecto es de 582 horas.

7.2 Presupuesto

En el presupuesto se van a detallar los costes en primer lugar del personal utilizado, a continuación del hardware que se necesita y por último del coste del software necesitado.

7.2.1 Coste del personal

A continuación se va a detallar el coste del personal que es necesario para llevar a cabo el proyecto. Estos roles son desarrollados por el autor de este proyecto.

Tabla 6 : Coste del personal detallado.

	Rol	Coste(€/hora)	Duración	Coste
Fase 0	Analista	40	24 horas	960€
Fase 1	Analista	40	60 horas	2400€
Fase 2	Analista	40	24 horas	960€
Fase 3	Programador	20	30 horas	600€
Fase 4	Programador	20	150 horas	3000€
Fase 5	Ingeniero de Pruebas	30	120 horas	3600€
Fase 6	Programador	20	24 horas	480€
Fase 7	Ingeniero de Pruebas	30	90 horas	2700€
Fase 8	Analista	40	60 horas	2400€
TOTAL			582 horas	17100€

La dirección de este proyecto se ha llevado a cabo por la Dra. Sara Pino Povedano, que empleó aproximadamente 200 horas en llevar esta tarea, puesto que sus honorarios ascienden a 40€/hora, el coste total de la dirección es de 8000€.

Por lo que el coste total del personal de este proyecto es de **25100€**.

7.2.2 Coste del hardware

Ahora se van a detallar los costes producidos por los elementos hardware que se han necesitado en este proyecto. Estos costes vienen detallados en la Tabla 7.

Tabla 7 : Coste detallado del material hardware.

Concepto	Coste(€)	% Uso dedicado al proyecto	Dedicación (meses)	Período de depreciación (meses)	Coste imputable ^(*) (€)
Ordenador sobremesa	700	100	6	60	70
Router ADSL	50	100	6	60	5
USB	10	100	6	60	1
TOTAL					76 €

(*) Fórmula de depreciación:

$$\frac{A}{B} \times C \times D$$

A = nº de meses desde la fecha de facturación en que el equipo es utilizado

B = periodo de depreciación (60 meses)

C = coste del equipo (sin IVA)

D = % del uso que se dedica al proyecto

7.2.3 Coste del software

En este apartado se van a desarrollar los costes relacionados con el software necesario para el desarrollo del proyecto.

En el ordenador personal se encuentra instalado Windows 7 professional, cuya licencia fue obtenida de forma gratuita a través de la universidad. Para la realización de esta memoria ha sido necesario utilizar Microsoft Office 2010 Hogar y Estudiante por el que hubo que abonar la licencia. Para la implementación del sistema ha sido necesaria la utilización del programa MATLAB 2010 que también supone un coste puesto que el programa no es gratuito. Los costes nombrados vienen detallados en Tabla 8.

Tabla 8 : Coste detallado del material software.

Concepto	Coste (€)
Licencia Windows 7 Professional	0
Licencia Microsoft Office 2010 Hogar y Estudiante	89
Programa MATLAB 2010	1500
TOTAL	1589 €

7.2.4 Coste del Material de Oficina

El coste del material de oficina que se ha utilizado para este proyecto se estima en aproximadamente **150 €**.

7.2.5 Costes indirectos

Los costes indirectos son la partida destinada a gastos como pueden ser la luz, el agua, etc. Para este proyecto se asigna un **20%** de costes indirectos sobre el coste total del proyecto.

7.2.6 Coste Total

En este apartado se va a calcular el coste total del proyecto, apoyándonos en los costes obtenidos en los apartados anteriores. Esto se puede ver detallado en la Figura 25.



UNIVERSIDAD CARLOS III DE MADRID
Escuela Politécnica Superior

PRESUPUESTO DE PROYECTO

1.- Autor: Marta Salas Costumero

2.- Departamento: Teoría de la Señal y Comun

3.- Descripción del Proyecto:

- Título: **Técnicas de Detección para Aplicaciones de Localización en Redes de Sensores**
- Duración (meses): **6**
Tasa de costes indirectos: **20%**

4.- Presupuesto total del Proyecto (valores en Euros):

26.915,00 Euros

5.- Desglose presupuestario (costes directos)

PERSONAL

Apellidos y nombre	N.I.F. (no rellenar - solo a título informativo)	Categoría	Dedicación (hombres mes) ^{a)}	Coste hombre mes	Coste (Euro)	Firma de conformidad
Salas Costumero, Marta		Analista	1,28	5.250,00	6.720,00	
Salas Costumero, Marta		Programador	1,554	2.625,48	4.080,00	
Salas Costumero, Marta		Ingeniero Pruebas	1,6	3.937,50	6.300,00	
Pino Povedano, Sara					0,00	
					0,00	
Hombres mes 4,434				Total	17.100,00	

^{a)} 1 Hombre mes = 131,25 horas. Máximo anual de dedicación de 12 hombres mes (1575 horas)
Máximo anual para PDI de la Universidad Carlos III de Madrid de 8,8 hombres mes (1.155 horas)

EQUIPOS

Descripción	Coste (Euro)	% Uso dedicado proyecto	Dedicación (meses)	Periodo de depreciación	Coste imputable ^{a)}
Ordenador sobremesa	700,00	100	6	60	70,00
Router ADSL	50,00	100	6	60	5,00
USB	10,00	100	6	60	1,00
Total					76,00

^{a)} Fórmula de cálculo de la Amortización:

$$\frac{A}{B} \times C \times D$$

A = nº de meses desde la fecha de facturación en que el equipo es utilizado

B = periodo de depreciación (60 meses)

C = coste del equipo (sin IVA)

D = % del uso que se dedica al proyecto (habitualmente 100%)

SUBCONTRATACIÓN DE TAREAS

Descripción	Empresa	Coste imputable
Total		0,00

OTROS COSTES DIRECTOS DEL PROYECTO^{a)}

Descripción	Empresa	Costes imputable
Licencia Microsoft Office 2010		
Hogar y Estudiante		89,00
Programa MATLAB 2010		1.500,00
Material Oficina		150,00
Total		1.739,00

^{a)} Este capítulo de gastos incluye todos los gastos no contemplados en los conceptos anteriores, por ejemplo: fungible, viajes y dietas, otros,...

6.- Resumen de costes

Presupuesto Costes Totales	Presupuesto Costes Totales
Personal	17.100
Amortización	76
Subcontratación de tareas	0
Costes de funcionamiento	1.739
Costes indirectos	3.783
Total	22.698

Figura 25 : Presupuesto total.

CAPÍTULO 7 PRESUPUESTO

Se concluye finalmente que el presupuesto total de este proyecto asciende a la cantidad de **22.698€**.

Leganés a 12 de Diciembre de 2011

El ingeniero proyectista

Fdo. Marta Salas Costumero

Referencias

- [1] C. Intanagonwiwat, R. Govindan, d. Estrin, J. Heidemann, and F. Silva, "Directed diffusion for wireless sensor networking," IEEE/ACM Trans. Networking, vol. 11, no. 1, pp. 2-16, 2003.
- [2] Juan Liu, James Reich y F.Zhao, "Collaborative In-Network Processing for Target Tracking".EURASIP Journal on Applied Signal Processing, no. 4, pp. 378-391, 2003.
- [3] Z.Chair and P.K Varshney, "Optimal data fusion in multiple sensor detection systems," IEEE Trans, Aerosp. Electron. Syst., vol.22, no.1,pp. 98-101,1986.
- [4] E-Drakopoulos and C.-C. Lee, "Optimum multisensor fusion of correlated local decisions," IEEE, Aerosp. Electron. Syst., vol.27, no.4, pp. 593-606, 1991.
- [5] Niu, Ruixin; Moore, Michael; and Klamer, Dale, "Decision Fusion in a Wireless Sensor Network with a Large Number of Sensors" (2004). Electrical Engineering and Computer Science. Paper 82.
- [6] Ruixin Niu y Pramod K. Varshney, "Distributed Detection and Fusion in a Large Wireless Sensor Network of Random Size", EURASIP Journal on Wireless Communications and Networking 2005:4, 462–472.

- [7] Chio Cho Nayibe, Tibaduiza Burgos Diego Alexander , Aparicio Zafra Laura Cristina y Caro Ortiz Luis Miguel, “ Redes de Sensores Inalámbricos”. Universidad Autónoma de Bucaramanga. [<http://catic.unab.edu.co/2congresomecatronica/images/docum/9.pdf>]. Última visita: 2 de Febrero de 2011.
- [8] N. Bulusu, D. Estrin, L. Girod, J. Heidemann, Scalable coordination for wireless sensor networks: self-configuring localization systems, International Symposium on Communication Theory and Applications (ISCTA 2001), Ambleside, UK, July 2001.
- [9] Xiaohong Sheng and Yu-Hen Hu, "Collaborative Source Localization in Wireless Sensor Network System," IEEE Globecom, 2003.
- [10] I.A. Essa, Ubiquitous sensing for smart and aware environments, IEEE Personal Communications (October 2000) 47–49.
- [11] A. Cerpa, J. Elson, M. Hamilton, J. Zhao, Habitat monitoring: application driver for wireless communications technology, ACM SIGCOMM'2000, Costa Rica, April 2001.
- [12] Rocío Arroyo-Valles, Sara Pino-Povedano y Jesús Cid-Sueiro, “Selective Forwarding for Energy-efficient Target Tracking in Sensor Networks”, IEEE Journal on selected areas in communications.
- [13] Wickens, T.D. (2002). “Elementary signal detection theory”. Oxford: Oxford University Press.
- [14] S. Cho, A. Chandrakasan, Energy-efficient protocols for low duty cycle wireless microsensor, Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Maui, HI Vol. 2 (2000), p. 10.
- [15] D. Estrin, R. Govindan, J. Heidemann, Embedding the Internet, Communication ACM 43 (2000) 38–41.
- [16] C. Herring, S. Kaplan, Component-based software systems for smart environments, IEEE Personal Communications, October 2000, pp. 60–61.
- [17] J.M. Kahn, R.H. Katz, K.S.J. Pister, Next century challenges: mobile networking for smart dust, Proceedings of the ACM MobiCom'99, Washington, USA, 1999, pp. 271–278.
- [18] Feng Zhao y Leonidas J. Guibas, “Wireless Sensor Networks: An Information Processing Approach”. Elsevier . 2004.
- [19] F. Gustafsson, F. Gunnarsson, N. Bergman, U. Forssell, J. Jansson, R. Karlsson, and P.-J. Nordlund, “Particle filters for positioning, navigation, and tracking,” Signal Processing, IEEE Transactions on, vol. 50, no. 2, pp. 425 –437, feb 2002.

REFERENCIAS

- [20] N. Noury, T. Herve, V. Rialle, G. Virone, E. Mercier, G. Morey, A. Moro, T. Porcheron, Monitoring behavior in home using a smart fall sensor, IEEE-EMBS Special Topic Conference on Microtechnologies in Medicine and Biology, October 2000, pp. 607–610.
- [21] E.M. Petriu, N.D. Georganas, D.C. Petriu, D. Makrakis, V.Z. Groza, Sensor-based information appliances, IEEE Instrumentation and Measurement Magazine (December 2000) 31–35.
- [22] J.F.G De Freitas A. Doucet and N.J.Gordon. Sequential Monte Carlo Methods in Practice. New York, 2001.
- [23] N. Gordon M. Arulampalam, S. Maskell and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. IEEE Signal Processing, 50:174_188, Feb 2002.